Perturbations in Boolean Networks

as Model of Gene Regulatory Dynamics

Von der Fakultät für Mathematik und Informatik
der Universität Leipzig
angenommene

Dissertation

zur Erlangung des akademischen Grades

Doctor rerum naturalium
(Dr. rer. nat.)

im Fachgebiet

Informatik

vorgelegt

von M. Sc. Physics Fakhteh Ghanbarnejad

geboren am 2. September 1984 in Teheran

Die Annahme der Dissertation wurde empfohlen von:

1. Prof. Dr. Peter F. Stadler (Leipzig, Deutschland)
2. Prof. Dr. Olli Yli-Harja (Tampere, Finnland)

Die Verleihung des akademischen Grades erfolgt mit Bestehen der Verteitigung am 14.09.2012 mit dem Gesamtprädikat *magna cum laude*.

To my parents

Contents

Abstract

Boolean networks are coarse-grained models of the regulatory dynamics that controls the survival and proliferation of a living cell. The dynamics is time- and state-discrete. This Boolean abstraction assumes that small differences in concentration levels are irrelevant; and the binary distinction of a low or a high concentration of each biomolecule is sufficient to capture the dynamics. In this work, we briefly introduce the gene regulatory models, where with the advent of system-specific Boolean models, new conceptual questions and analytical and numerical challenges arise. In particular, the response of the system to external intervention presents a novel area of research.

Thus first we investigate how to quantify a node's individual impact on dynamics in a more detailed manner than an averaging against all eligible perturbations. Since each node now represents a specific biochemical entity, it is the subject of our interest. The prediction of nodes' dynamical impacts from the model may be compared to the empirical data from biological experiments.

Then we develop a hybrid model that incorporates both continuous and discrete random Boolean networks to compare the reaction of the dynamics against small as well as flip perturbations in different regimes. We show that the chaotic behaviour disappears in high sensitive Boolean ensembles with respect to continuous small fluctuations in contrast to the flipping.

Finally, we discuss the role of distributing delays in stabilizing of the Boolean dynamics against noise. These studies are expected to trigger additional experiments and lead to improvement of models in gene regulatory dynamics.

Acknowledgments

I would like so simply just say, thank you!

Konstantin Klemm
Peter F. Stadler
My beloved mother, father and Hadi
Mona D.
Jens, Petra
The beerinformaticians
Atefeh, Fatemeh S., Florian, Ghazaleh, Gunnar, Ida, Mahboubeh, Maribel, Mina, Mohammad, Mona H., Nadjieh, Nasim, Roghayeh, Sima, Shahin, Wolfgang, ...
All my teachers and professors
...

... and all the people who have influenced my thinking

CHAPTER 1

Introduction

> "Once upon a time, a little boy was looking for something under a street light. "What are you looking for?" asked an old man passing by. "My key" he replied. "Where have you lost it?!" He asked while looking around confusedly. "In that dark area over there" said the boy pointing to an area meters away. "So why are looking for it here?!!" said the old man surprisedly. "Because there is light over here!" replied the boy."
>
> *What scientists do* based on a tale I heard in a *Critical Phenomena* class taught by Prof. Shahin Rohani

The functioning of organisms on the molecular level is a research topic of increasing attention. Survival and reproduction requires an autonomous regulation of chemical concentrations in the living cell. In this chapter we open our discussions with an introduction to gene regulatory dynamics, the applied and developed mathematical formalisms and computational aspects.

1.1 Gene Regulatory Dynamics

The first discovery of a gene regulation system was done in 1961 by Jacob and Monod [1]. They found that some enzymes involved in lactose metabolism are expressed by the genome of E. coli only in the presence of lactose and absence of glucose. The importance of gene regulations for living cells in order to increase the versatility and adaptability of an organism has since triggered a central attention of researchers in this area.

For introducing the genetic regulatory systems, Hidde De Jing starts his review paper [2] by the following words:

> "In order to understand the functioning of organisms on the molecular level, we need to know which genes are expressed, when and where in the organism, and to which extent. The regulation of gene expression is achieved through genetic regulatory systems structured by networks of interactions between DNA, RNA, proteins, and small molecules."

He continues to make a general picture of such systems by addressing gene expression as follows:

> "Gene expression is a complex process regulated at several stages in the synthesis of proteins [3]. Apart from the regulation of DNA transcription, the best-studied form of regulation, the expression of a gene may be controlled during RNA processing and transport (in eukaryotes), RNA translation, and the posttranslational modification of proteins. The degradation of proteins and intermediate RNA products can also be regulated in the cell. The proteins fulfilling the above regulatory functions are produced by other genes. This gives rise to genetic regulatory systems structured by networks of regulatory interactions between DNA, RNA, proteins, and small molecules. An example of a simple regulatory network, involving three genes that code for proteins inhibiting the expression of other genes, is shown in Fig. 1.1. Proteins B and C independently repress gene a by binding to different regulatory sites of the gene, while A and D interact to form a heterodimer that binds to a regulatory site of gene b. Binding of the repressor proteins prevents RNA polymerase from transcribing the genes downstream."

Certain reasons make the understanding and modeling of the regulatory systems a hot research topic. The first one is their entanglement with the complexity concept. After finding the lack of correlation between genome size and complexity, known as the C-value paradox [4], as well as between the number of genes and complexity, known as the G-value paradox [5], the paradigm of complexity of organisms has been shifting to the complexity of the regulatory control structures where for example various combinations of 20000 genes might allow up to 10^{6000} different gene expression patterns [6]. The second outstanding point which is motivating scientists to study such dynamics is their relevance to the evolution which occurs with genetic variability and could be understood in the context of regulations [7]. Recent works point to the key roles that regulatory changes could play in evolutionary processes e.g. in primate evolution [8] or the evolution of human and chimpanzee transcriptome [9]. And the last but not the least motivation to investigate these systems is also their possible relevance to the cause and treatment of cancer [10, 11, 12].

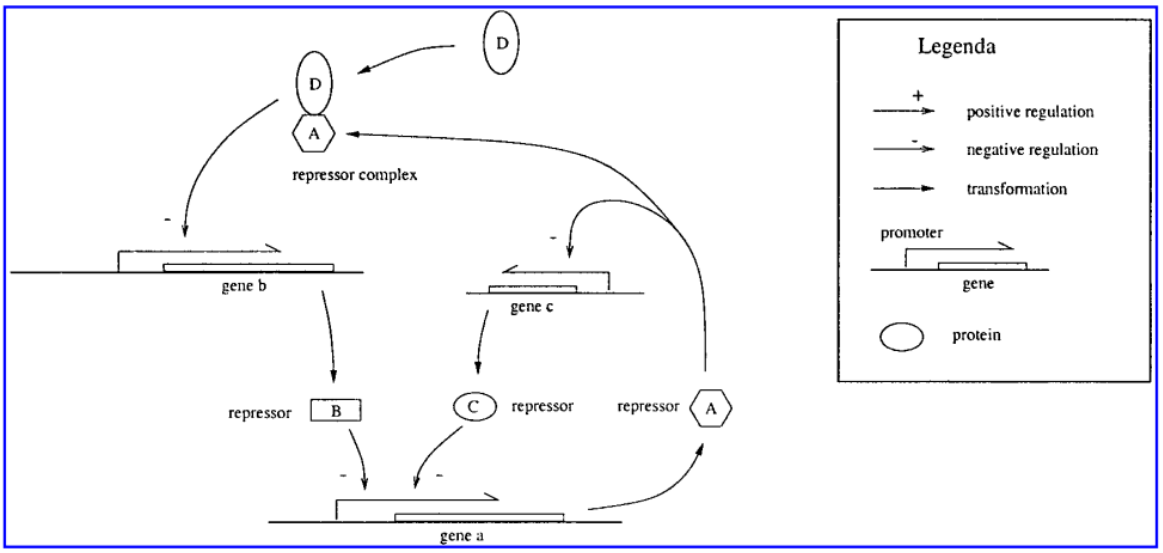


Figure 1.1: Example of a genetic regulatory system, consisting of a network of three genes a, b, and c, repressor proteins A, B, C, and D, and their mutual interactions. The figure distinguishes several types of interaction. The figure and the caption is adopted from Ref. [2].

1.2 Mathematical Formalisms

Different mathematical formalisms and numerical methods have been used for modelling such regulatory dynamics. These models deal with different levels of information, complexity and computational expenses. Classifying available models in the different levels (see figure 1.2) Bornholdt tries to develop a general intuition that how to build a fit model for large genetic networks [13]; What details can be ignored without being unfaithful to the general behaviour of the system? Which element can be considered as the building block of the model? Which parameters are ignorable while the others are vital in our modelling? These are questions that he addresses to review the history of modeling of gene regulatory systems and extrapolate the best option for the levels not reached yet (right column in Fig. 1.2) [13].

Here we focus on two main classes of models: *small genetic circuit* and *mid-size genetic network* models as shown schematically in the figure 1.2, center left column and center right column respectively. In this point of view, various mathematical approaches have been developed, from discrete to continuous methods, from deterministic to stochastic techniques, from static to dynamical models, from detailed and fine to coarse grained perspectives; Let us take a more detailed look at the well-structured review paper [2] and rewrite it in our own words. Here we re-sort some of the reviewed models according to their dynamical state details whether they are continuous or discrete as listed in the table 1.1, since we need this perspective in later chapters[1].

[1]We are not discussing all of these models in details, As we merely want to create a general image from the

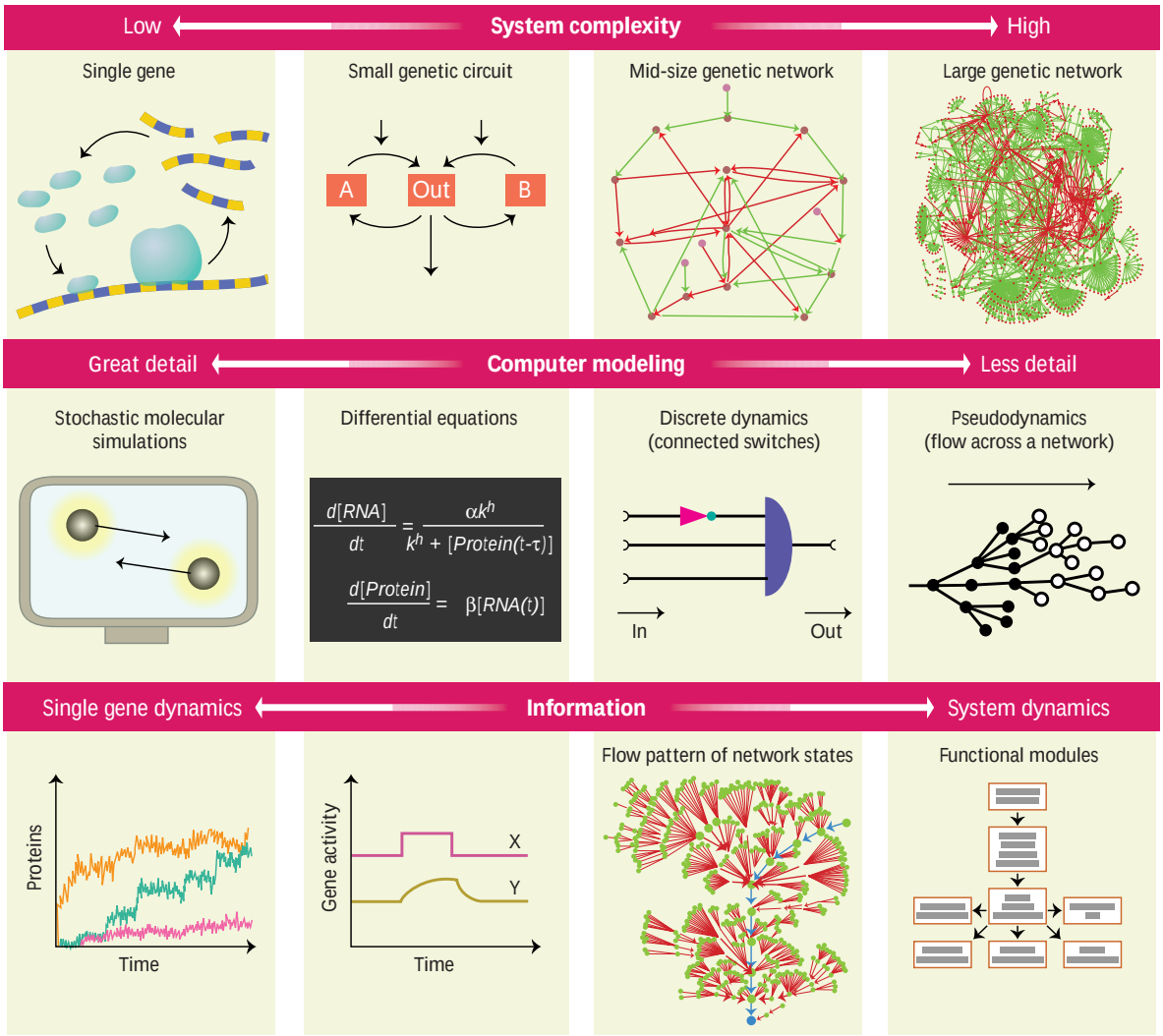


Figure 1.2: The different levels of mathematical modelling for gene regulatory networks. While stochastic simulations can present single genes in molecular detail (left column), modelling circuits of genes with differential equation is more applicative (center left column). By turning to mid-sized genetic circuits switch-like ON/OFF dynamical models can faithfully reproduce the general behaviour of the system (center right column). We should take lessons from percolating flows across a network structure in terms of how to model a larger genetic network (right column). The figure and the caption is adopted from Ref. [13].

Table 1.1: Continuous and discrete models of gene regulatory systems. Table is taken from Ref. [2].

Models	Discrete(d), Continuous(c)
Nonlinear Differential Equations (NLDE)	c
Piecewise-Linear Differential Equations (PLDE)	c
Directed Graphs (DG)	–[1]
Boolean Networks (BN)	d
Generalized Logical Networks (GLN)	d
Stochastic Master Equations (SME)	d

[1] This is a static model, so there is no dynamics to be continuous or discrete.

1.2.1 Continuous Models

The common approach to model a typical dynamical system in science and engineering is *ordinary differential equations* (ODE). They were applied to study biological regulatory systems as well, where solutions represent the continuous values of concentrations x_i of genes, RNAs, proteins or other types of molecules. In other words, the models give us the rate of productions of the biochemical elements as a function of time which is given by the general equation 1.1.

$$\frac{dx_i(t)}{dt} = f_i(x(t)) \quad i \in \{1, ..., N\} \tag{1.1}$$

when $f_i : R^N \rightarrow R$ is a suitable linear, nonlinear or piece-wise linear function. Dealing with circuits of genes, this kind of modelling is more practical than detailed single genes models [13].

Nonlinear Ordinary Differential Equations In most cases, the best fit choice in Eq. 1.1 is a nonlinear function (*nonlinear differential equations* - NLDE); and sometimes equations include some time delays representing the time that the system needs to pass the information or materials from one element to the other by transcriptions, translations, diffusions or other transporting processes in the system (*delay differential equations* - DDEs). E.g. the oscillations of protein Hes1 were reproduced by Eq. 1.2. Connection of this protein to cell differentiation made it an important and interesting case to study [14].

applied and developed mathematical models in the field of regulatory systems in the direction needed in the later discussions. Please read the reference [2] and its references for more models, technical details and examples.

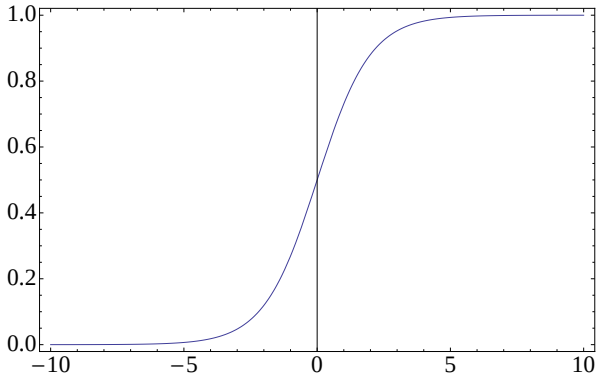


Figure 1.3: An example of s-shaped curve. The plot represents following function, $x = \frac{1}{1+exp^{-t}}$ $t \in (-10, 10)$

$$\begin{aligned} \frac{d[mRNA]}{d[t]} &= \frac{\alpha k^h}{k^h + [Hes1(t-\tau)]^h} - \frac{[mRNA(t)]}{\tau_{rna}} \\ \frac{d[Hes1]}{d[t]} &= \beta[mRNA(t)] - \frac{[Hes1(t)]}{\tau_{hes1}} \end{aligned} \tag{1.2}$$

Where the parameters $\tau, \alpha, k, h, \tau_{hes1}, \tau_{rna}$ denote production delay, production rates, characteristic concentration, binding threshold h which is the Hill coefficient and characteristic degradation times respectively.

Piecewise-Linear Differential Equations To simplify qualitative analysis, let us consider the switching behaviour of the biological components from lowest concentration values to the highest rates or vice versa by continuous s-shaped curves which are known as sigmoid functions (one of them is shown in Fig. 1.3). The mathematical properties of this type of functions and the mapped biological dynamics were studied in Ref. [15]. So this simplification provides *piecewise-linear differential equation* (PLDE) models formulated in the general form 1.3; which make us ready to work on later abstract discrete models such as *Boolean networks* and *generalized logical networks*.

$$\frac{dx_i(t)}{dt} = g_i(x(t)) - \gamma_i x_i \quad i \in \{1, ..., N\}, \quad \gamma_i > 0 \tag{1.3}$$

where x_i plays the role of the production value of gene i and γ denotes its degradation rate. Here the f function in Eq. 1.1 is replaced by the piece-wise linear function $g_i(x(t)) - \gamma_i x_i$.

$$\begin{aligned} \frac{d[x_j(t)]}{d[t]} &= g_j(x_{j-1}(t-\tau)) - x_j(t) \quad \text{with} \\ g_j(x_i) &= \eta_j \frac{1 + d_i^j x_i^\nu}{\tau_{1+b_i^j x_i^\nu}} \end{aligned} \tag{1.4}$$

Equation 1.4 denotes a theoretically-studied example of circuits with PLDE [16].

1.2.2 Discrete Models

Despite the fact that the first group of models studied here have been in good agreement with detailed experimental databases, some problems should be taken in the account. On the one hand, the point is that usually it's not possible to find an exact, analytical solution to the equations, due to the presence of the nonlinear terms. Thus the numerical methods give us a chance to make an approximation to the behaviour of the dynamics, find the steady states and test the stability of them.

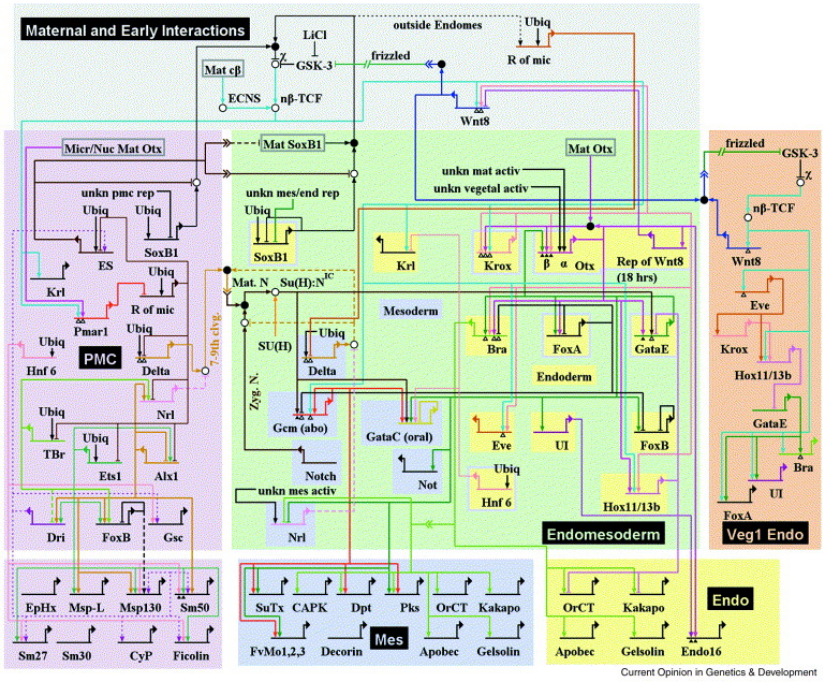


Figure 1.4: An example of mid-size regulatory genetic network in the sea urchin embryos. See Ref. [17] for details and this figure.

On the other hand, one should note the large number of parameters needed to reproduce the dynamics of just one single gene, as seen in the above example of equation 1.2. The high computational costs associated with the modelling of larger regulatory systems (such as

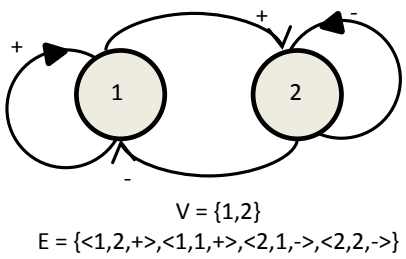


Figure 1.5: A simple directed graph. In the set E denotes that gene i activates gene j and $< i, j, - >$ denotes that i is inhibiting j.

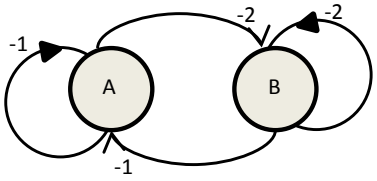
the one shown in figure 1.4) has led to the simplification of models. Therefore going from *small genetic circuit* to *mid-size genetic network*, a transition happened in modelling, from *continuous* to *discrete* models. Moreover, some studies have shown that such models are quite stable to represent and predict the overall behaviour of the system against variations of kinetic parameters over several orders of magnitude [14]. This was yet another important fact that encouraged modellers to transition to another level of modelling and focus their attention to the structure of the interactions more than precise values of the parameters; Thus messy parametric models have been replaced by models with much fewer parameters or even parameterless models.

Directed Graphs Mapping the interactions between the elements of the regulatory system like genes or proteins to a *directed graph* (DG) provides us with a static model, where edges tell us which element influences which other elements. Type of information that edges can carry is whether one element is activating or inhibiting the others, which is shown by a simple sign $+/-$, respectively. Figure 1.5 shows an example of a directed graph G defined by $< V, E >$; When V presents set of nodes and E shows set of interactions. These graphs can be obtained by various approaches. One might directly compose the graph from databases or use reverse engineering methods.

Analyzing the mathematical characters of the mapped graph provides some information about the original biological data. For instance, cycles in the graph indicate feedback relations. The average and the distribution of the interactions give us an estimation of the complexity of the system and so on.

Boolean Networks *Boolean networks* (BN)[18, 19, 20, 21, 22, 23] is another framework for modeling regulatory systems, especially for precise sequence control as observed in morphogenesis [24] and cell cycle dynamics [25] but also in the regulation of the metabolism [26].

An N-dimensional Boolean map $f : \{0, 1\}^N \rightarrow \{0, 1\}^N$ gives rise to a time-discrete dynam-



x_1x_2	f_A	f_B
00	1	2
01	1	2
02	1	1
10	1	2
11	0	2
12	0	1
20	1	0
21	0	0
22	0	0

Figure 1.6: An example of generalized logical networks. Edge labels show which thresholds should be taken, the signs indicate inhibitors and excitatories and specified logical rules are written in the table.

ics

$$x(t+1) = f(x(t)) \tag{1.5}$$

with $x = (x_1, \ldots, x_N)$ being a Boolean state vector (bit string) of N entries. Such a map is equivalent to a *Boolean network*. When f is pictured as a network, a node corresponds to a coordinate i of the Boolean state vector and a directed edge $j \to i$ (from node j to node i) is present if the Boolean function f_i explicitly depends on the j-th coordinate. The main idea behind these types of modeling is that we can ignore the intermediate expression levels of genes and consider them as occupying either of two states of active (on, 1) or inactive (off, 0) at any given time. See chapter 2 for further detailed discussions.

Generalized Logical Networks To generalize Boolean networks, one can consider variables with more than two values. The extension has been developed in different directions; However we only describe the main idea of generalization by the following formalism [27]. Let us consider $p+1$ state values. In this case, the continuous concentration rates are mapped to the abstract state values, $x_i \in \{0, 1, 2, ..., p\}$ according to the defined thresholds, $\theta_i^1 < \theta_i^2 < ... < \theta_i^p$ written in equation 1.6.

$$\begin{aligned} x_i &= 0, \quad \text{if} \quad x_i < \theta_i^1 \\ x_i &= 1, \quad \text{if} \quad \theta_i^1 \le x_i < \theta_i^2 \\ &... \\ x_i &= p, \quad \text{if} \quad x_i \ge \theta_i^p \end{aligned} \tag{1.6}$$

and dynamics still follows the equation 1.5, but f is a more complicated logical rule, $f : \{0, 1, ..., p\}^N \to \{0, 1, ..., p\}^N$. Figure 1.6 indicates a simple example of *generalized logical*

networks (GLN). The graph has two nodes which are regulated by the other node states, first input, and their own states, second input, in the last time step. The system has three thresholds which means states have three possible values, $x_i \in \{0, 1, 2\}$; So the state space includes $(p+1)^N = 3^2 = 9$ states and logical functions, generalized Boolean functions, can be chosen from $(p+1)^{(p+1)^N} = 3^{3^2} = 19683$ possibilities and should be consistent with imposed restrictions of the thresholds. Edges are labelled with the rank number of thresholds, while the signs show whether the regulations are excitatory or inhibitory. e.g. -2 means that node A is an inhibitor of node B and influences it above its second threshold. The functions, $f_A = f(x_B, x_A), f_B = f(x_A, x_B)$ are specified in the right hand table of figure.

Since the dynamics is deterministic and state space is finite, as are the Boolean networks, any initial condition ends to a *logical steady state or cycle*; These steady states could correspond to the patterns of gene expression [28].

Stochastic Master Equations Since some assumptions of the continuous and deterministic approaches may be questionable according to the nature of the regulatory phenomena, discrete and stochastic models were proposed as another alternative (see Ref. [2] and its references). Eq. 1.7 presents a general form of such models; where the discrete variable X refers to the molecules' states, $p(X, t)$ indicates the probability that at time t the cell contains a given amount of different molecules of $X_1, X_2, \ldots$; m is the number of the chemical reactions making the dynamics, β_j is the probability that reaction j bring the system from another state to the current state X while $\alpha_j p(X, t)$ is the probability that reaction j is in the state X at time t. Thus the *stochastic master equation* (SME) 1.7 gives the time evolution of the joint probability distribution $p(X, t)$.

$$\frac{\partial p(X,t)}{\partial t} = \sum_{j=1}^{m} [\beta_j - \alpha_j p(X,t)] \tag{1.7}$$

1.3 Computational Aspects

Let us summarize what we said so far. We introduced the biological phenomena which we are trying to understand and model; Then we reviewed the most widely used and developed mathematical formalisms mapped to them. Now let us speak a bit about the limitations and problems in these two realms, i.e. experiment and theory. On the one hand, sometimes to setting up an experiment is not very easy because of various shortages such as lack of technological facilities, expenses, etc.; or there are some problems in the later stages such as storing of a huge body of empirical data and further analysing the stored databases statistically. On the other hand, mapping a mathematical formalism to the laboratory observations which is able to explain the phenomena in the language of precise numbers is not a very easy and simple task. Also whenever the mathematical model seems to fit the data, its predic-

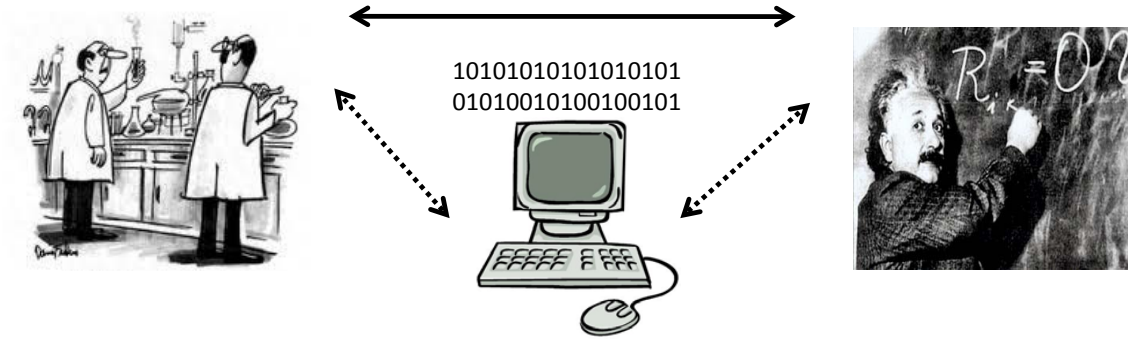


Figure 1.7: The computational bridge between experiment and theory, the external world and human internal world, in the natural sciences. In the last century scientists were equipped by computers to shorten the gap between laboratory and mind worlds.

tions should be tested; But how can we confirm the theory where experiment is not possible? Would that be the end of the story? Of course not! Here is where computational approaches come into play. Simulations try to provide the missing link between experiment and theory in all different levels of natural sciences, Fig. 1.7. Although computational methods struggle with their own limitations, borders, costs and difficulties, numerical results provided by them can make outlines for theoreticians as well as experimentalists in terms of where to search for what, thereby reducing the costs and speeding up the research projects to achieve the goals more efficiently. Our approach in this thesis is computational and aims at assessing the limitations and power of mathematical models mapping gene regulatory systems. From this point of view, we do specifically focus on Boolean dynamics and particularly perturbations on dynamics, since Boolean networks are good candidates for modelling regulatory dynamics and enable us to pass from one level of modelling to the larger system sizes [13].

Let us go back to our main concern, namely the study of gene regulatory systems, and revisit it now by a computational gloss to understand what makes Boolean networks specific in term of computations. We try to find similarities between the regulatory systems and a simple washing machine which we work with daily[2]. Thus there are two outstanding lessons from computer engineering which give us hints as to how to improve our methods. First, a washing machine has a designed control circuit which receives some input data like temperature, water level, switches, etc; and after analyzing it makes an output including a sequence of instructions to tune the temperature, water level and so on. Similarly we can reduce our model to a dynamics including a series of molecular activations which regulates the cell cycle elements such as cell size, temperature, stress, food supply and so on, in response to the received input information e.g. damage, food sources and supply, light and any other internal or external data or stimuli. Second, we need to know how to build up this sequence in

[2]This example is adopted from Ref. [29]

mathematical language. Which details must be considered in the dynamics and which are not necessary? Since the machine alphabet is binary, a good option is a binary dynamics which can represent the dynamical actions (or reactions) by a sequence of the involved gene states (0s and 1s) ignoring the switching details. These models are the ones which we know as Boolean networks.

1.4 Structure of the Thesis

What motivates us to study the models properties, discussed in section 1.2, is a better mathematical understanding of gene regulatory systems explained in Sec. 1.1; Finally in the section 1.3 we tried to address, very briefly, the main idea behind the computational aspects which could help us improve the mathematical frameworks and enable us to find out the limitations and problems of the developed models.

In this thesis, we study the role of perturbations in Boolean networks as a model of gene regulatory dynamics. Hence first we introduce the mathematical framework of Boolean networks in chapter 2. Then we review in chapter 3 the nature of noise in regulatory systems and studies of different types of fluctuations in mathematical formalisms.

In chapter 4, we make centrality measures based on topological as well as dynamical characters of the Boolean networks to rank nodes with respect to their capability to spread perturbations on dynamics. The chapter is based on the following preprint paper:

- Ghanbarnejad F, Klemm K (2011). **Impact of individual nodes in Boolean network dynamics**, *Europhysics letters (resubmitted after minor revision, 2012)*, in preprint *arXiv:1111.5334v1*.

In chapter 5, we investigate the dynamical resilience of random Boolean networks against flip perturbations and small perturbations. The chapter is based on the following publication:

- Ghanbarnejad F, Klemm K (2011). **Stability of Boolean and continuous dynamics**, *Phys. Rev. Lett. 107. 188701.*

In chapter 6, we study spreading of flip perturbations on dynamics updated synchronously according to the nodes' flat distributed delays.

Finally we summarize our take home messages in chapter 7. The appendix A presents some technical tricks in the simulations.

CHAPTER 2

Boolean Networks: Mathematical Framework

"Mathematics is the art of giving the same name to different things."

Jules Henri Poincare (1854-1912)

A Boolean network is a state- and time-discrete dynamical system. The dynamics is defined by an iteration

$$x(t+1) = f(x(t)) \tag{2.1}$$

with N Boolean dynamical variables written as a binary vector $x(t) \in \{0,1\}^N$ at each time $t \in \mathbb{N} \cup \{0\}$. The mapping $f : \{0,1\}^N \to \{0,1\}^N$ is typically sparse: calculating the state $x_j(t+1)$ requires knowledge of the state $x_i(t)$ for a few ($\ll N$) indices i at the previous time step. When the system is pictured as a directed network, the nodes $\{1, 2, \ldots, N\}$ carry the dynamic variables $x_1, x_2, \ldots, x_N$ interacting along a relatively small number of directed arcs. Subindices address components of a vector such that x_j is the Boolean state of node j and f_j is its Boolean function.

In order to formalize and quantify these ideas, we consider the x_i-dependence of f as the mapping

$$\partial^{(i)} f_j(x) = \begin{cases} 1 & \text{if } f_j(x) \neq f_j(x^{\updownarrow i}) \\ 0 & \text{otherwise} \end{cases} \tag{2.2}$$

This is the Boolean analogue of the usual partial derivative of a function, using $x^{\updownarrow i}$ to denote state vector x with its i-th entry negated. Note that $\partial^{(i)} f$ also maps from $\{0,1\}^N$ to $\{0,1\}^N$.

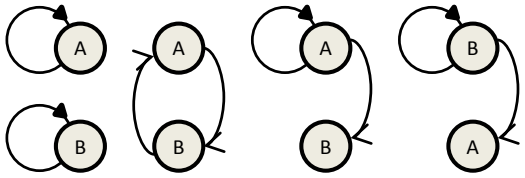


Figure 2.1: A $(2,1)$ Boolean network can have 4 different topologies presented above.

By averaging $\partial^{(i)} f$ over all states with equal weight, the activity of i on j is obtained as

$$\alpha_{ij}(f) = 2^{-N} \sum_{x \in \{0,1\}^N} \partial^{(i)} f_j(x) \tag{2.3}$$

The *activity* $\alpha_{ij}(f)$ is the probability that a perturbation (negation of state) at node i causes a perturbation at node j in the subsequent time step, assuming that all 2^N state vectors occur with equal probability. The *sensitivity* of the Boolean function f_i is the sum of its incoming activities,

$$s_i(f) = \sum_{j=1}^{N} \alpha_{ji} \ . \tag{2.4}$$

Likewise, we define the *strength* of f_i as the sum of outgoing activities [32]

$$\sigma_i(f) = \sum_{j=1}^{N} \alpha_{ij} \ . \tag{2.5}$$

Any random Boolean network is unique by its specified topology (Sec. 2.1) and dynamical rules (Sec. 2.2).

2.1 Graph

The directed network on the nodes $\{1, 2, \ldots, N\}$ obtained from f contains an arc from node i to node j if and only if $\alpha_{ij}(f) \neq 0$. The adjacency matrix A of the network has an entry $a_{ij} = 1$ if $\alpha_{ij}(f) \neq 0$ and $a_{ij} = 0$ otherwise.

An N-node network can have various topologies based on different in-degree and out-degree probability distributions. Restricting all nodes to receive K inputs leads to the flat probability distribution function, $P_{in} = \frac{K}{N}$, and therefore a Poisson distribution of out-degree, $P_{out}(k) = \frac{K^k}{k!} e^{-K}$, in the limit of $N \to \infty$ [21]. These Boolean networks are known as (N, K). Figure 2.1 presents 4 possible $(2,1)$ Boolean networks; Another alternative for in- or out-degree distribution is *power law*, $P(k) = \frac{k^{-\gamma}}{\sum_{i=k_{min}}^{k_{max}} k^{-\gamma}}$, which is well-known for biological networks [33]

Table 2.1: Different possible Boolean functions corresponding to 1 input(up) and 2 inputs(down). The functions can be classified in 3 groups: a) reversible like f_1, f_2, F_6, F_9 where the output changes by changing an input b) frozen like f_0, f_3, F_0, F_{15} where the output is constant independent of its inputs c) canalizing like the rest of the functions which have a fixed output corresponding to the value of canalizing input regardless of the other inputs

In	f_0	f_1	f_2	f_3
0	0	1	0	1
1	0	0	1	1

In	F_0	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}
00	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
01	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
10	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
11	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

and can be generated by different methods [34].

2.2 Dynamics

In this mathematical framework, each node's state can be 1 or 0; Therefore the system has 2^N configurations which build the *state space*. A specific dynamics makes a trajectory in the state space and can be distinguished by its chosen Boolean functions and the updating scheme.

2.2.1 Boolean Functions

A Boolean function is a mapping based on logical calculations from K Boolean inputs to a Boolean output. Thus there are 2^{2^K} Boolean functions corresponding to K inputs. Table 2.1 shows possible Boolean functions of one input and 2 inputs as well. The functions can also be chosen by different probability distributions. Drossel has listed 5 choices of Boolean functions distributions in her review paper [21].

2.2.2 Updating Schemes

The last piece of information we need to run the dynamics is to know how to update nodes by their chosen Boolean rules, whether parallel or not. Figure 2.2 presents a comprehensive classification of random Boolean networks according to the updating schemes. The most commonly used updating schemes in the Boolean literature are listed below.

Deterministic Synchronous (SRBN): Nodes are updated in order, e.g. from lowest to highest i based on the nodes' states in the previous time step.

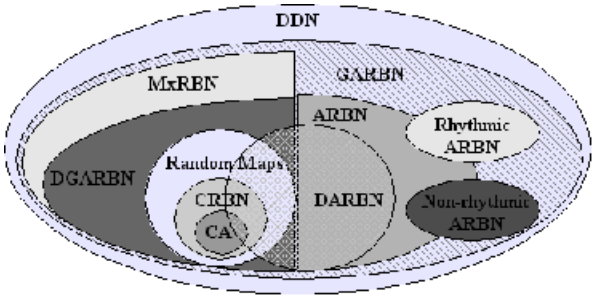


Figure 2.2: Classification of random Boolean networks, according to their updating schemes. The figure is taken from Ref. [35]. For definitions, details and discussions about updating schemes read the references [35, 36].

Asynchronous (ARBN): Nodes are updated one by one in a random order.

Deterministic Asynchronous (DARBN): Each node has two randomly generated parameters P_i and Q_i which are fixed during the entire process. Nodes are updated whenever $t \equiv Q_i \pmod{P_i}$; where t presents the time step. If more than one node satisfies the condition, all such nodes are updated synchronously [35].

In what follows, whenever we use *update* we mean *synchronous update*, unless otherwise stated.

2.3 Attractors

The long-term behaviour of Boolean dynamics is determined by attractors. These are minimal ergodic sets in the state space defined as follows.

Under synchronous update, an attractor of length l is a sequence of states $x(0), x(1), \ldots x(l-1)$ such that $f(x(t)) = x([t+1] \mod l)$ for all $t \in \{0, \ldots, l\}$.

Under asynchronous update, an attractor is a strongly connected component of the state transition graph, having all state vectors as its nodes [37].

Before moving on, we need to define some further terminology. An attractor with length 1 is a *fixed point*. The set of states which end at an attractor including the attractor's states is called the *basin of attractions*. And those states which are not on the attractor cycle but in the basin are called *transients*. Figures 2.3 and 2.4 illustrate these concepts with simple Boolean networks;

A quick review of the Boolean network literature reveals that better computational methods such as attractor-finding algorithms[1], the development of computers with larger memories,

[1]The algorithm applied in this dissertation to find Boolean attractors is explained in appendix A.1.

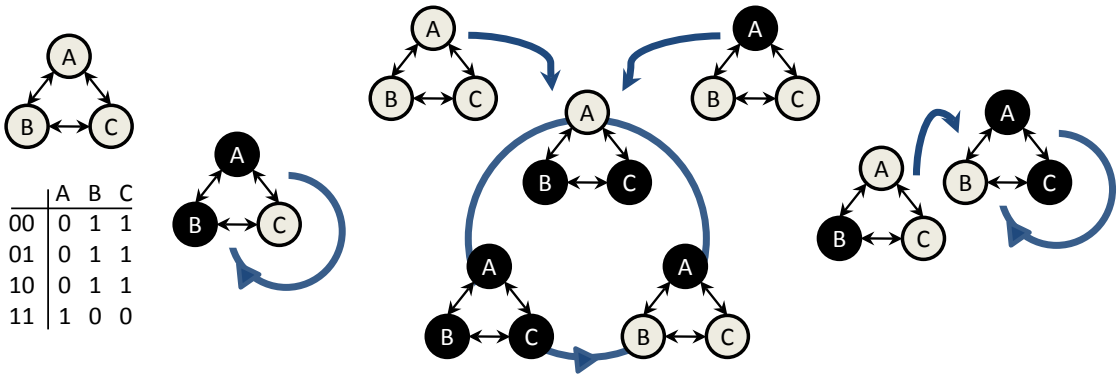


Figure 2.3: A $(3, 2)$ Boolean network (left up) updated by the following truth table (left down) and its corresponding state space (right). Black nodes are off(0) and grays are on(1). The dynamics has three attractors, two of which are fixed points. The largest basin in the state space includes 5 states, 2 of which are transients and the attractor length is 3.

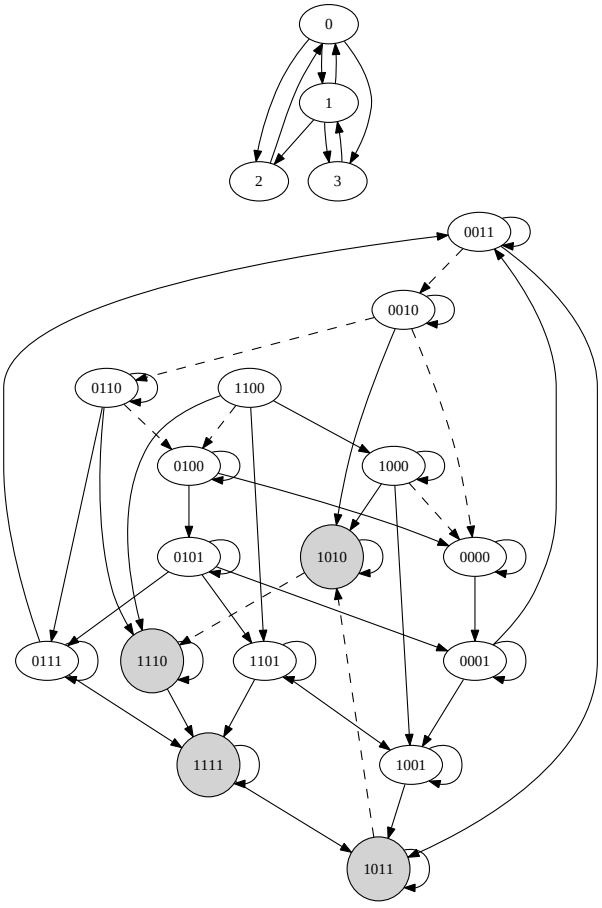


Figure 2.4: A $(4, 2)$ Boolean network and its state space due to an asynchronous updating fashion. Dashed lines indicate applied operators $U_{i+/-}$. Nodes $0, 1, 2, 3$ are from right to left. Circles are the attractor-states. The Figure is taken from [38]. See more technical details in the same reference.

as well as analytical arguments have led to a better knowledge and estimation of the mean number of attractors, the mean length of attractors and the distribution of attractors with different lengths as well. For instance the mean number of attractors of critical random Boolean networks (see section 2.5 for definition of criticality) versus the number of nodes has been estimated sublinear [39], linear [40], superlinear [41][2] and superpolynomial [42] so far.

2.4 Ensembles of Random Boolean Networks

Let us make a simple example; Applying 4 possible Boolean functions (Table 2.1) as the updating rule of each node in 4 different topologies of $(2, 1)$ (Figure 2.1) gives us 32 distinct Boolean networks which can be randomly chosen by determinate weights, probability distributions listed in Ref. [21] or other alternatives, and make an ensemble of random Boolean networks.

An ensemble of random Boolean networks [21] is defined by the number of nodes N, the number of inputs K of each node, and the probability distribution of Boolean functions $\pi(f)$. In this dissertation, the $\pi(f)$ is taken as a maximum entropy ensemble $\pi_\lambda(f) \propto \exp(\lambda s(f))$ under a given average sensitivity $\langle s \rangle$ (Eq. 2.4) with a free parameter λ.

A purely theoretical branch of studies is devoted to randomly constructed Boolean networks [43, 21] and strives to elucidate generic features of Boolean dynamics. From the perspective of statistical mechanics, averaged macroscopic quantities in the limit of large system size are described in dependence of ensemble parameters such as the probability distribution of the employed Boolean functions [44, 45] and the degree distributions of the networks [46, 47]. The number of attractors [48, 41, 42, 49] and the stability under perturbations [50, 46, 51, 52, 53, 54, 31] have been investigated.

2.5 Critical Boolean Networks

Here, the underlying fundamental result is a transition between convergent (stable) and divergent (unstable) dynamics when the input sensitivity (Eq. 2.4) of the Boolean functions passes a critical value [50, 55]. The phase transition shown in Fig. 2.5 has been obtained with different numerical or statistical analyses listed briefly following[3].

A: Dynamics of Hamming distance Given a random Boolean network, two identical copies are initialized with states that differ in some nodes, the number of nodes of difference is known as the *Hamming distance d*. Plotting the different initialized Hamming distance networks evolved in one time step gives us the *Derrida plot*, $\frac{d(t+1)}{d(t)} = \langle s \rangle$ [50]; where $\langle s \rangle$ is the average sensitivity of the generated Boolean ensemble. Figure 2.6 shows how the behavior of the

[2] an analytic calculation
[3] Three first listed are discussed with more details in [21]

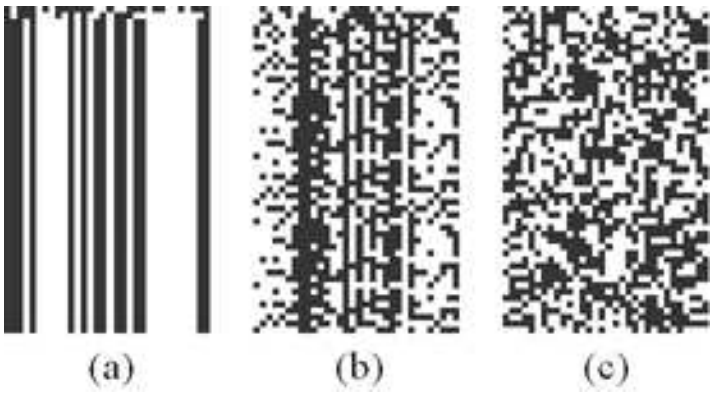


Figure 2.5: Presentation of three phases a) ordered(K=1), b) critical(K=2) and c) chaotic(K=5) through temporal evolution of dynamics for N=32. Squares show the state of nodes. The dynamics starts from the top and proceeds downwards. Figure is reprinted from [35].

dynamics changes in the critical point. Drossel has discussed properties of the networks in the subcritial ($\langle s \rangle < 1$), critical ($\langle s \rangle_c = 1$) and supercritial ($\langle s \rangle > 1$) regimes. She has also pointed out the values of the critical point based on Boolean network parameters for different distributions of Boolean functions [21].

But let us now make an estimation of the critical K_c calculated under assumptions of *annealed approximation* by Derrida and Pomeau [50]. Consider two replicas of the same network in which one system is in state x, the other in state y. Thus, the normalized Hamming distance is easily calculated as follows: $d = N^{-1} \sum_{i=1}^{N} |x_i - y_i|$. Tracking the time evolution of d, we have an iterative map $d(t+1) = 2p(1-p)[1-(1-d(t))^K]$; where p denotes the fraction of 1 to 0 in the output of the Boolean function, and $1-(1-d(t))^K$ is the probability that a given node is connected to at least one node j with $y_j \neq x_j$ and $p(1-p)+(1-p)p$ is the probability that the state of a node changes when at least one of its inputs changes.

$$f(d) = 2p(1-p)[1-(1-d(t))^K] \tag{2.6}$$

The iterated map in equation 2.6 has a trivial fixed point $d^* = 0$ whose stability is determined by its derivative in Eq. 2.7.

$$f'(d) = 2p(1-p)K(1-d(t))^{K-1} \tag{2.7}$$

f' is a strictly decreasing function and f has at most two fixed points. At the fixed point corresponding to $d^* = 0$, $f'(0)$ is equal to $2p(1-p)K$ and it is stable if $f'(d) < 1$ while it's unstable if $f'(d) > 1$. It means that $2Kp(1-p) = 1$ determines the critical boundaries which are shown in the Fig. 2.7 [43, 46].

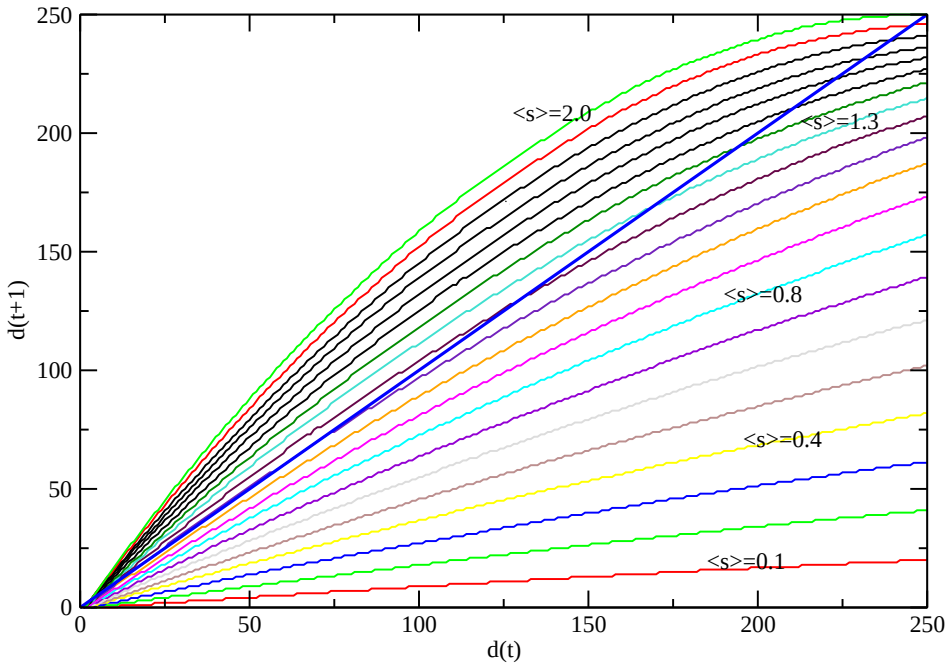


Figure 2.6: Derrida plot for maximum entropy ensembles of Boolean networks $(500, 2)$ (see Sec. 6.1 for more details). Average sensitivity $\langle s \rangle$ is changing from 0 to 2, down to up curves. Each point is an average over 1000 data.

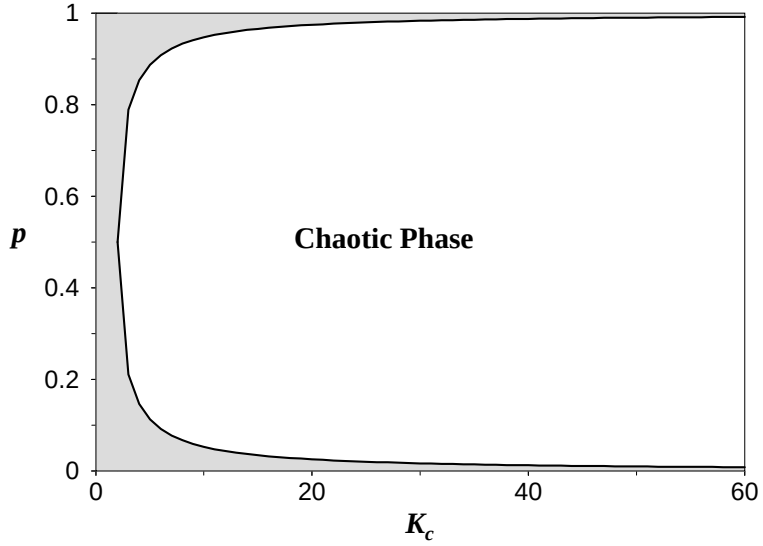


Figure 2.7: Due to annealed approximation, $2Kp(1-p) = 1$ determines the critical boundaries between order and chaotic phases. The chart is adopted from Ref. [46].

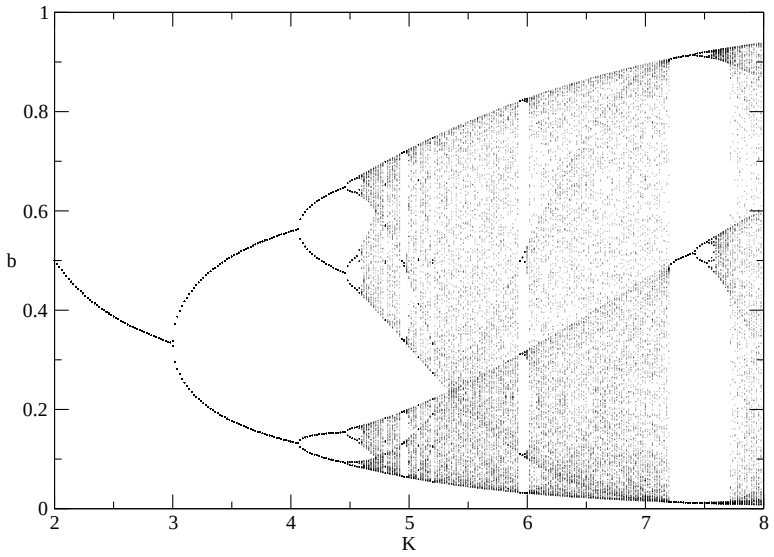


Figure 2.8: Representation of phase transition according to fraction of *on* nodes. Figure is taken from Ref. [21].

B: Proportion of 1s and 0s The time evolution of the fraction of *on* nodes, b_t by *annealed approximation* gives us an iterative map

$$b_{t+1} = b_t^K + (1 - b_t)^K \tag{2.8}$$

where K shows the in-degree of Boolean networks. Plotting b versus continuous K in Fig. 2.8 shows that it changes its behaviour from a stable fixed point to a periodic phase and then ends at a chaotic regime (see Ref. [21] for the calculations). The analysis went further to introduce an order parameter based on the Jacobian matrix [56]. And recently the method of fraction of *on* nodes, was also used to check the stability behaviour of the largest (so far) mapped Boolean dynamics to empirical data [22].

C: Probability of stable dynamics against damages Let two initialized copies of a random Boolean network which differ at one node state run in parallel and observe whether or not they eventually reach the same trajectory. Probability that the perturbations die out in a random generated ensemble represents the phase transition at critical point (see Fig. 2.9). Stability of random Boolean dynamics against flip perturbation and this phase transition is debated in details in chapter 5.

D: Lyapunov exponent It was calculated as order parameter through different methods and represented the second order phase transition at the critical point ($K = 2$) in agreement with other analyses [56]. Another attempt has been made to show chaotic behavior of a non-clocking regulatory dynamics with respect to Lyapunov exponent too; Zhang et. al

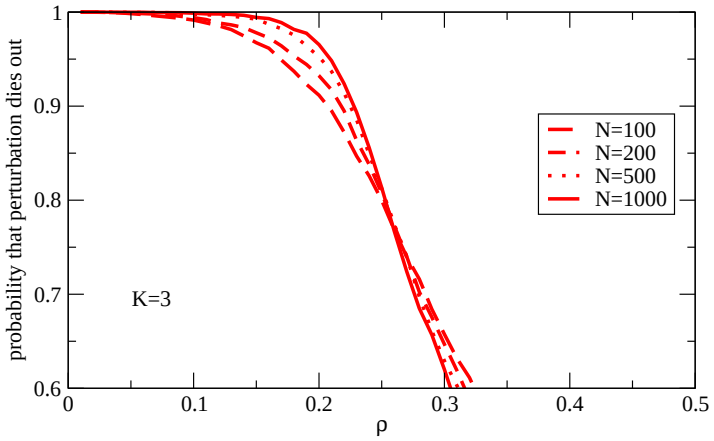


Figure 2.9: Stable dynamics transit to instability by increasing the control parameter ρ.

have applied a nonlinear time series analysis introduced by [57] to estimate the largest Lyapunov exponent of a $(3, 2)$ Boolean dynamics updated autonomously and the corresponding electronic logical gates data [58].

To close this section, let us keep in mind two points about the importance of critical Boolean networks. First of all, Kauffman believes that optimal regulatory functioning requires networks to be at the critical point. Since in the ordered phase, dynamics is frozen and can not respond to stimuli and in the chaotic phase, any small perturbation forces the system into remote part of the phase space. Thus evolution happens at the *edge of chaos* i.e. the critical regime [48]. Secondly, damage spreading in a critical random Boolean network leads to a power law distribution of cluster size affected by perturbations. This is closely related to critical percolation which has been studied theoretically in more depth [21].

2.6 Boolean vs. Continuous Dynamics

Using binary (on/off) concentrations as an idealization, Boolean dynamics directly implements the logical skeleton of regulation. Values of system parameters such as binding constants, production and degradation rates, etc., are not needed. This abstraction simplifies computation and analytical treatment. Boolean networks have been extracted directly from the literature [22, 59] of known biochemical interactions or obtained by discretization of differential equation models [60]. Known state sequences and responses of several systems have been faithfully reproduced by the discrete models [24, 25].

Despite these benefits, modelers do not employ Boolean dynamics as widely as ordinary or delay differential equations. The latter are embedded in an established framework for state-*continuous* dynamical systems [61] which itself builds on the mathematical foundations of linear algebra and infinitesimal calculus. In particular, the definition of *stability* of a solution under *small* perturbations is based on the consideration of infinitesimally small neighborhoods

Table 2.2: The comparison of a Boolean attractor (Fig. 2.3) vs. continuous counterpart dynamics (Fig. 2.10 (b)) according to the switching events, where $\uparrow i(\downarrow i)$ represents that node i is switching on (off). *Error time* is a metric that shows whether switching sequences of two dynamics in a rescaled mapped time overlap completely or differ. In the first case, it's equal to summation of all switches happened in the dynamics. But in the latter, it's less; e.g. 12 in our example, which means that after 12 switching events, the first difference in the switching sequence appears (see appendix A.3).

discrete time step[a]	Boolean switching sequence	Continuous switching sequence
1	$\uparrow 2, \uparrow 3$	$\uparrow 2, \uparrow 3$
2	$\uparrow 1, \downarrow 2, \downarrow 3$	$\downarrow 2, \downarrow 3, \uparrow 1$
3	$\downarrow 1$	$\downarrow 1$
4	$\uparrow 2, \uparrow 3$	$\uparrow 2, \uparrow 3$
5	$\uparrow 1, \downarrow 2, \downarrow 3$	$\downarrow 2, \downarrow 3, \uparrow 1$
6	$\downarrow 1$	$\uparrow 3$

[a] As it was explained in Eq. 2.10, continuous time series can be mapped to the discrete time step by time rescaling.

in the state space. Stability checks on the solutions of the dynamical equations are a salient part of mathematical modeling. Unstable solutions are not expected to be observed in a real-world system. We discuss the concept of stability in detail in chapter 5.

Let us now define a continuous dynamics whose discretization readily leads to the Boolean map in Eq. 1.5. Taking values $y_i(t) \in [0, 1]$, $i \in \{1, \ldots, N\}$, $t \in \mathbb{R}$, the states evolve according to the delay differential equation

$$\dot{y_i}(t+1) = \alpha \, \mathrm{sgn}(\tilde{f}(y(t)) - y_i(t+1)) \tag{2.9}$$

with α an inverse time constant. For large α, this is essentially Boolean dynamics with fast but continuous switching between the saturation values. The simplest choice is $\tilde{f} = f \circ \Theta$ with Θ the component-wise step function, $\Theta_i(y) = 1$ if $y_i \geq 1/2$ and $\Theta_i(y) = 0$ otherwise. This choice of continuous dynamics is in close correspondence with the discrete dynamics in the following sense. Suppose $x(0), x(1), x(2), \ldots$ is a solution of Eq. 1.5. Let $y(t)$ be a solution of Eq. 2.9 such that there is a time interval $[t_1, t_2]$ with $y(s) = x(0)$ for all $s \in [t_1, t_2]$. Then for all future times $t \in \mathbb{N}$ and all $s \in [t_1, t_2]$

$$x(t) = y(\beta t + s) \tag{2.10}$$

with $\beta = 1 + 1/(2\alpha)$.

The closest resemblance between Boolean and continuous dynamics is obtained when choosing the same initial condition, that is $y(s) = x(0)$ for all $s \in [-1, 0]$. Table 2.2 clarifies how the Boolean attractor of Fig. 2.3 can be compared with the corresponding continuous dynamics

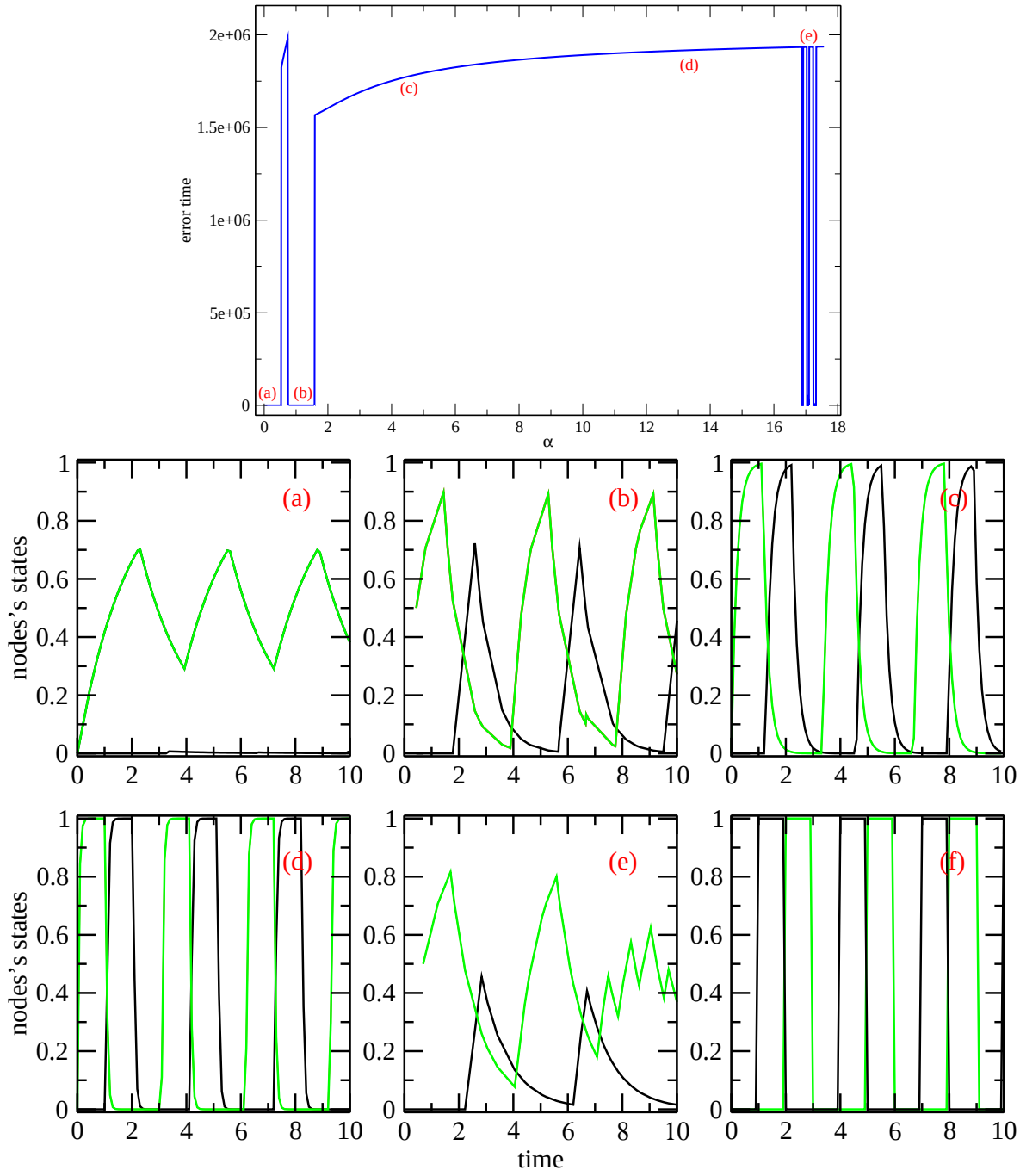


Figure 2.10: Transition of Boolean dynamics, the largest attractor in Fig. 2.3, from continuous to discrete with respect to the α parameter. The continuous dynamics satisfies $\dot{x_i}(t) = \alpha[\Theta(h_i(t-1)) - x_i(t)]$; where Θ is the step function with threshold at $\frac{1}{2}$ with $h_i(t) = ax_{i,in1}(t)x_{i,in2}(t) + b_1x_{i,in1}(t) + b_2x_{i,in2}(t) + c$; where, for instance, a node performing $x_{in1} \wedge x_{in2}$ has $a = 1$ and $b_1 = b_2 = c = 0$. For other Boolean functions see table 5.1. Continuous and discrete dynamics run parallel till $t < 1000$. The stored switching events are compared in discrete time steps to find out when the two dynamics differ for the first time. *Error time* curve, explained in table 2.2, shows where the two dynamics are well-matched. While (f) presents an ideal discretesized Boolean dynamics; (a), (b), (c) and (d) show how the dynamics gets closer to the ideal case for larger α. (e) denotes the precision problem which we faced when using threshold methods to speed up the simulations (see appendix A.3).

(see table 2.2 and appendix A.3). Figure 2.10 illustrates how the transition from a continuous dynamics to the discrete happens by changing the α.

A similar correspondence between Boolean maps and ordinary differential equations has been studied earlier neglecting transmission delay [62] or implementing more complicated differential equations [63, 16, 64, 65] compared to Eq. 2.9.

2.7 Empirical Boolean Networks

In recent years, the theory of random ensembles has been complemented by case studies showing that suitably constructed Boolean networks capture the behaviour of empirical regulatory systems [66, 24, 25, 22, 20]. These system-specific Boolean networks are obtained by compiling biochemical interactions from the literature [59], by discretizing existing models of differential equations [60], or by inference from data by a dedicated algorithm [67].

Let us introduce four empirical networks in the following, where the first three ones are small and the last one is thus far the biggest well-matched empirical-theoretical Boolean dynamics.

Mammalian cell cycle: Fauer et al. have introduced a cell cycle model by a Boolean network; The network has 10 nodes representing proteins which inhabit or activate each other by a Boolean function [68]. The authors try three different updating schemes, both synchronous and asynchronous updating as well as the mixed version, to assess the corresponding attractors and interpret them as the counterpart biological process. Figure 2.11 displays the regulatory graph and its nodes' logical operators.

Yeast cell cycle: Li et al. have modeled a very vital biological process, cell division, by a relatively small network with 11 proteins [25]. Constructing an interaction network with threshold Boolean functions 2.11, the model can predict the four-phase cycle of the cell division: G1, S, G2 and M in yeast. Figure 2.12 shows the graph which comprises 34 different interactions, 15 of which are activating and 19 are inhibiting including self-couplings; and the corresponding state space.

$$x_i(t+1) = \begin{cases} 1 & \text{if } \sum_j w_{ij} x_j(t) > 0, \\ 0 & \text{if } \sum_j w_{ij} x_j(t) < 0, \\ x_i(t) & \text{else.} \end{cases} \tag{2.11}$$

Th network: Mendoza et al. have developed a method to model the regulatory networks and find their steady states from discrete to continuous dynamics [70]; Then as an example, they have modeled the regulatory network (Fig. 2.13) which controls the diffrentiation process of T

$$
\begin{array}{ll}
CycD & CycD \\
\text{Rb} & (\overline{CycD} \wedge \overline{CycE} \wedge \overline{CycA} \wedge \overline{CycB}) \\
 & \quad \vee (p27 \wedge \overline{CycD} \wedge \overline{CycB}) \\
\text{E2F} & (\overline{Rb} \wedge \overline{CycA} \wedge \overline{CycB}) \vee (p27 \wedge \overline{Rb} \wedge \overline{CycB}) \\
\text{CycE} & (E2F \wedge \overline{Rb}) \\
\text{CycA} & (E2F \wedge \overline{Rb} \wedge \overline{Cdc20} \wedge \overline{(Cdh1 \wedge Ubc)}) \\
 & \quad \vee (CycA \wedge \overline{Rb} \wedge \overline{Cdc20} \wedge \overline{(Cdh1 \wedge Ubc)}) \\
\text{p27} & (\overline{CycD} \wedge \overline{CycE} \wedge \overline{CycA} \wedge \overline{CycB}) \\
 & \quad \vee (p27 \wedge \overline{(CycE \wedge CycA)} \wedge \overline{CycB} \wedge \overline{CycD}) \\
\text{Cdc20} & CycB \\
\text{Cdh1} & (\overline{CycA1} \wedge \overline{CycB}) \vee (Cdc20) \vee (p27 \wedge \overline{CycB}) \\
\text{UbcH10} & (\overline{Cdh1}) \vee (Cdh1 \wedge Ubc \\
 & \quad \wedge (Cdc20 \vee CycA \vee CycB)) \\
\text{CycB} & (\overline{Cdc20} \wedge \overline{Cdh1})
\end{array}
$$

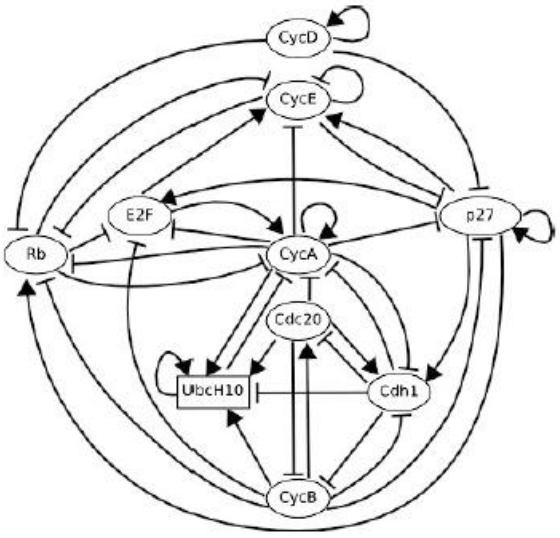


Figure 2.11: Boolean network for the mammalian cell cycle network. Right side shows the logical regulatory graph. Each node represents the activity of a key regulatory element, whereas the edges represent cross-regulations. Blunt arrows stand for inhibitory effects, normal arrows for activations. And left-hand side shows the logical rules underlying the definition of the logical parameters associated with the right graph. See Ref. [68] for more details and this figure.

helper cells and has 23 proteins with Boolean interaction rules satisfying the set of equations 2.12.

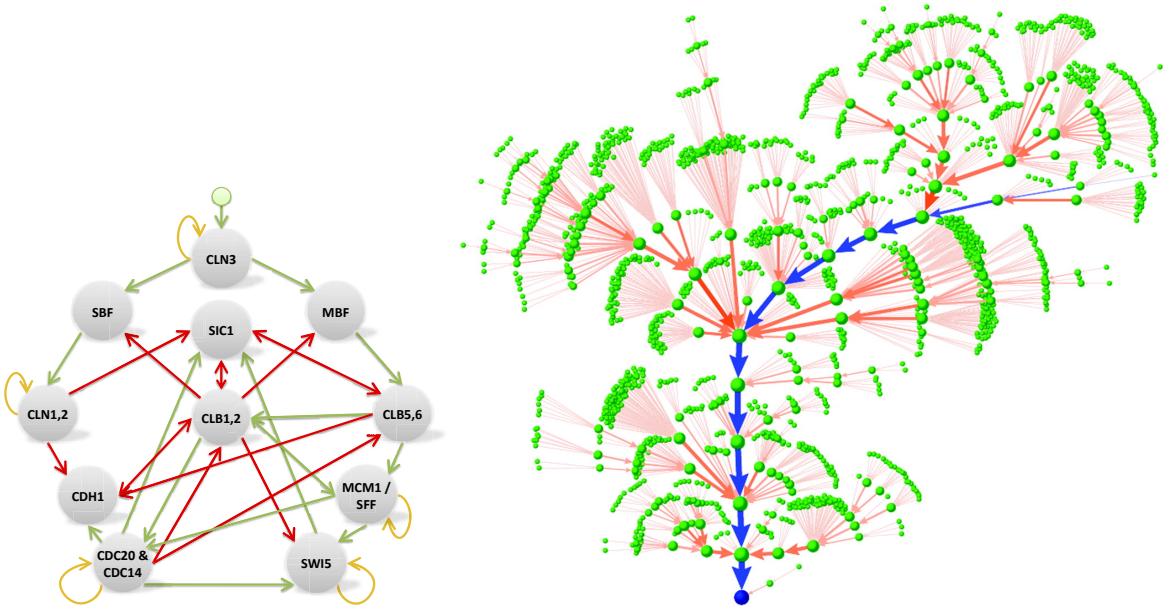


Figure 2.12: The regulatory network of the yeast cell cycle (left). Green arrows are activating regulators when red are self-couplings and yellow arrows are inhibitors. The state space of the dynamics is represented in right-hand. See Ref. [25, 69] for more details and these figures.

$$
\begin{aligned}
GATA3(t+1) &= (GATA3(t) \vee STAT6(t)) \wedge \neg(T-bet(t)) \qquad (2.12)\\
IFN-\beta R(t+1) &= IFN-\beta(t)\\
IFN-\gamma(t+1) &= (IRAK(t) \vee NFAT(t) \vee STAT-4(t) \vee T-bet(t)) \wedge \neg(STAT3(t))\\
IFN-\gamma R(t+1) &= IFN-\gamma(t)\\
IL-10(t+1) &= GATA3(t)\\
IL-10R(t+1) &= IL-10(t)\\
IL-12R(t+1) &= IL-12(t)\\
IL-18R(t+1) &= IL-18(t) \wedge \neg(STAT6(t))\\
IL-4(t+1) &= GATA3(t) \wedge \neg(STAT1(t))\\
IL-4R(t+1) &= IL-4(t) \wedge \neg(SOCS1(t))\\
IRAK(t+1) &= IL-18R(t)\\
JAK1(t+1) &= IFN-\gamma R(t) \wedge \neg(SOCS1(t))\\
NFAT(t+1) &= TCR(t)\\
SOCS1(t+1) &= STAT1(t) \vee T-bet(t)\\
STAT1(t+1) &= IFN-\beta R(t) \vee JAK1(t)\\
STAT3(t+1) &= IL-10R(t)\\
STAT4(t+1) &= IL-12R(t) \wedge \neg(GATA3(t))\\
STAT6(t+1) &= IL-4R(t)\\
T-bet(t+1) &= (STAT1(t) \vee T-bet(t)) \wedge \neg(GATA3(t))
\end{aligned}
$$

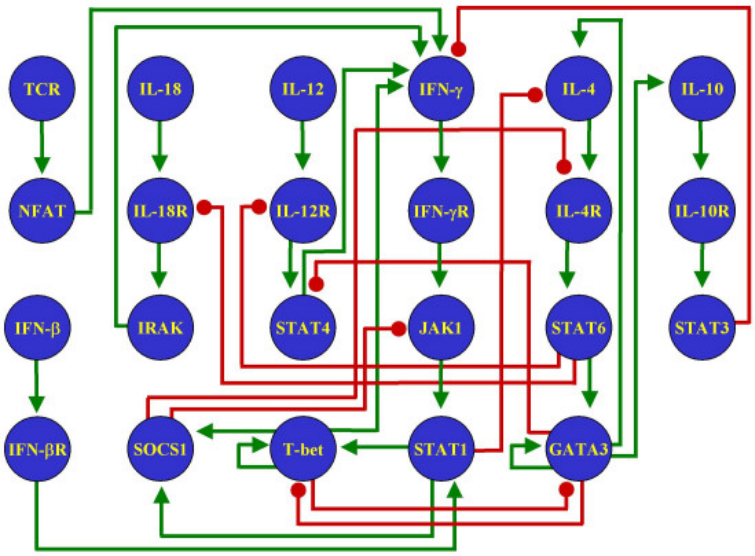


Figure 2.13: The Th network. The regulatory network that controls the differentiation process of T helper cells. Positive regulatory interactions are in green and negative interactions in red. The Figure is adopted from the Fig. 2 in Ref. [70].

Fibroblast signal transduction dynamics: Helikar et al. describe signal transduction in fibroblasts with a detailed Boolean network [22, 71], with 139 nodes including 59 self-couplings. This network as an interesting case because of its size and because of its large number of intertwined feedback loops of various lengths l (see also Fig. 1 in Ref. [71]). We quantify the abundance of feedback by the trace of the l-th power of the adjacency matrix A, finding $\mathrm{tr}(A) = 59$, $\mathrm{tr}(A^2)/2 = 568$, $\mathrm{tr}(A^3)/3 = 82455$ and $\mathrm{tr}(\mathrm{A}^4)/4 = 13921796$. The nodes fall into two classes. There are 9 *input nodes* with a self-coupling. Each of these applies the identity function to its own state, not receiving signals from any other node. These nodes provide constant but choosable input to the rest of the network. Each of the remaining 130 nodes receives an input from at least one other node in this set. We call these the *core nodes*. The in-degree of nodes varies from 1 to 14, the out-degree varies from 1 to 28.

CHAPTER 3

Noise in Gene Regulatory Dynamics

> "Could we not imagine that noise...is itself nothing more than the sum of a multitude of different sounds which are being heard simultaneously?"
>
> *Dictionnaire de Musique (1767)*
> Jean-Jacques Rousseau (1712-1778)

A challenging topic in modeling of genetic regulatory networks is on the one hand understanding the origin of the astonishing robustness against different types of noise observed in such biological phenomena. While on the other hand, the evolvability and evolutionary priority of the models in different regimes makes this subject more mysterious. Several analytical and numerical attempts have been established to study the role of noise in the steady states in continuous and discrete dynamics. Considering the nature of noise in the regulatory systems, here we make a brief review on the nature of noise, perturbations and the stability concept in mathematical frameworks.

3.1 The Nature of Noise

Experimental data indicate that dynamical biological processes including cell polarization, signal transduction and other vital dynamics for cells are taking place in highly noisy backgrounds. To create any kind of perturbation in the solutions to the mathematical models for further studies, one should consider the nature of noise existing in the real phenomena. Eldar and Elowitz have reviewed types of fluctuations as regards to gene expression as a central cellular function. The nature of noise is characterized by the following basic concepts.

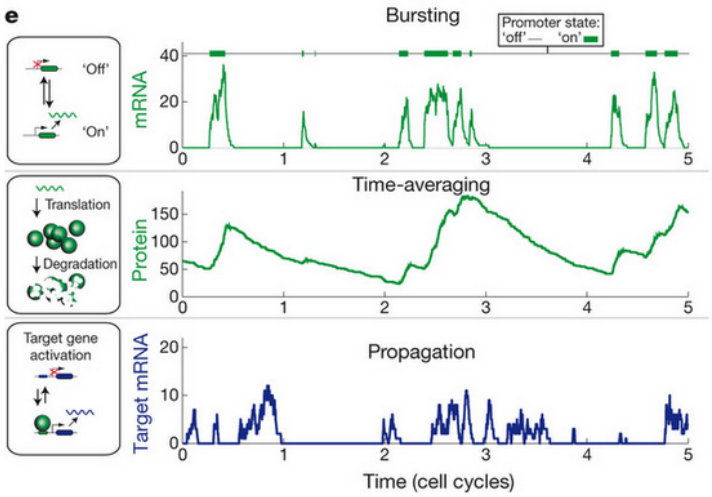


Figure 3.1: Characteristics of noise in the gene expression. The top chart shows intrinsic noise, while the bottom chart shows noise propagation, or extrinsic noise. The middle chart shows that the time interval between bursts are typically shorter than protein life. See the original figure in Ref. [72] for more details.

First, the protein concentrations which dictate the level of activity or inhibition of the genes have a very *bursty* character and can sometimes be very low (typically thousands, but sometimes hundreds or even tens at times [72]). Second, propagation of a fluctuations in one gene can affect the whole regulatory dynamics. Notice that the time interval between bursts are typically shorter than protein life time (Fig. 3.1, middle). Another notable point is that noise-generating bursts, or *intrinsic noise*, lead to uncorrelated fluctuations (Fig. 3.1, top) while correlated fluctuations present noise propagation, or *extrinsic noise* (Fig. 3.1, bottom). Although at the first glance, it seems that noise interferes with the dynamics, they went on to demonstrate the functional roles of noise in genetic circuits. Analyzing state switching, excitability and procrastination dynamics points to the usefulness of the noise on enabling physiological regulation mechanisms, permission of a wide range of probabilistic differentiation strategies and facilitation of the evolutionary adaptations [72]. Hereinafter we continue this chapter by focusing on perturbations applied in small and mid-size gene circuits testing the stability of the dynamics.

3.2 Mathematical formalisms and Fluctuations

Having the trajectories of dynamical systems based on any mathematical framework, an important question arises as to how the dynamics reacts in the long-term to small perturbations in initial conditions. Broadly defined, the dynamics is stable if small changes in the system lead to small variations in its behavior; otherwise it is unstable. To make the definition more

precise, one must specify the initial conditions which can possibly be perturbed, the perturbations and the measurement of variations in the dynamics. Here we take a fast look at the mathematical models, the different definitions of noise in these models, and the reaction of the dynamics against them.

Some works has been devoted to illustrating the behaviour of the continuous dynamics in response to external noise. These works also try to address other questions such as how positive as well as negative feedback loops play role in dynamical stability of the system, which conditions lead to stable periodicity or multistationarity. In continuous models (see section 1.2.1) often the stability of solutions of differential equations is tested by tracking trajectories of dynamical systems under *small perturbations* of initial conditions. Various criteria have been developed to measure stability or instability of the steady states. Sometimes the question is reducible to a well-studied problem involving eigenvalues of Jacobian matrices which is coming from a more general method known as Lyapunov functions. But let us now return to discrete models which are more practical for larger system sizes (see section 1.2.2) with specific focus on Boolean networks. Thus the following includes a couple of examples of these various approaches where the definition of noise is varied too.

- Functional reliability of information processing has been studied in the presence of *timing fluctuations*. Thus the resilience of the dynamics against this type of noise for feedback loops of two nodes, three nodes, and arbitrary number of nodes was investigated. It was shown that reliable three-node topologies take part in natural networks significantly more often than the randomized ones [73]. A later study has shown that the yeast cell cycle's control network [25] has surprisingly stable dynamics against this type of noise. Nevertheless, for making any biological interpretation, one should take into account that the different time scales presenting the different processes in the system were considered equal and many details were ignored to simplify the model [74].

- By defining two order parameters including the long-term average of the Hamming distance and the average frozenness as a function of noise strength, Peixoto and Drossel have studied the effect of noise on the random Boolean dynamics. While they have considered *noise as a probability p that a node does not follow its original Boolean function*, it was shown that the smooth transition from deterministic ($p = 0$) to fully stochastic ($p = 1/2$) dynamics takes place for sub- and critical networks ($K \leq 2$) where a first order transition happens at ($p = 0$) for super-critical networks ($K > 2$). Most of the numerical results of this investigation were also derived analytically based on annealed approximations [75].

- Considering *expression bias as the probability p that denotes the presence of the off outputs in the Boolean update rules*, as well as in-/out-degree correlation, Pomerance et. al. have linearized the dynamics solution around the fixed point for small perturbations

and governed a stability test for the steady states respecting to the largest eigenvalue of the modified adjacency matrix [12].

- The reaction of the system against two types of noise applied to the *update schedule* and *update rule* was investigated. The dynamical robustness was defined as the chance that the dynamics come back to the reliable trajectory by *flipping a single node*, in which the attractors of the Boolean dynamics which present high robustness against *changing or randomizing the updating schedules* are named reliable trajectories. Therefore it was found that many minimal networks are highly robust by applying the evolutionary algorithm. Here "minimality" means that the nodes' connectivity is not greater than what the dynamics needs to reach a desired reliable trajectory [54].

- The role of a specific class of Boolean functions on dynamical robustness has been well studied too. Thus the Derrida plots and the percolation simulations denoted that networks generated from mostly canalizing functions tend to show ordered behaviour against the *flip perturbations* as noise [51].

3.3 Discussion

Since some models, for instance the signal transduction network [22], have shown a higher dynamical robustness than expected to be seen, this interesting topic is still an open question for the modellers. What can represent different existing noise in the systems in mathematical language and is the dynamics robust or/and evolvable against them? Answering these questions becomes more crucial when mathematical formalisms vary over a broad range of continuous models 1.2.1 to discrete models 1.2.2 and deal with different levels of information, complexity and computational details (see Fig. 1.2).

In most studies, flip perturbations in random Boolean networks are commonly understood as noise. Flipping was considered as the equivalent of infinitesimal perturbations of the continuous counterparts of Boolean network models. This assumption could be questionable. In certain cases, the presence of a very small number of transcription factor molecules can increase the expression of a single gene by a factor of 10 or even 100. As a rule of thumb, the standard deviation of the concentration level of such proteins should be on the order of $N^{\frac{-1}{2}}$, where N is the number of proteins. More detailed estimations show however that for gene regulation the situation is far worse, and the standard deviation should scale with $N^{\frac{-1}{4}}$ (see Lestas et. al. [76]). Thus, for small enough values of N, as it is often the case, infinitesimal perturbations are much less realistic than *flip perturbations*. Of course, in the idealized, and very unrealistic, limit where N goes to infinity, *infinitesimal perturbations* should be the deciding factor for stability. Therefore in chapter 5 we make a technical comparison between the two types of perturbations, flip and small, in a hybrid model. The correspondence between

the two approaches can in principle show where the Boolean approximation breaks down in terms of stability.

CHAPTER 4

Dynamical Impact of Individual Nodes in Boolean Networks

"One can cut out a tooth but not an eye."

Persian proverb

The goal of this chapter is to establish a formal notion of node impact in Boolean dynamics and its relation to a node's topological position in the network. We perform a linear approximation of the long-term effect of a perturbation at a specific node i. We find that, in good approximation, the expected impact is monotonically related to the entry of i in the leading eigenvector of the adjacency matrix. When not only the network structure but also the Boolean functions are known, the estimate is improved by replacing the adjacency matrix with a weighted matrix of the activity values derived from the functions. The analytic approximations are validated by numerical studies of random Boolean networks and an empirical network from the literature[1].

4.1 Introduction

Considering some structural and dynamical characteristics of a complex network, several attempts have focused on finding a suitable quantity to measure the static as well as dynamic centrality of individuals. These measure a nodes importance or prominence in the network. The more central a node is in a network the more significant it is to aid in the spread of infection. Nodes have been ranked according to their degree, core, shell, betweenness,

[1] This chapter is adopted from [30].

closeness and eigenvector centrality most widely[2].

Klemm et al. have shown that *dynamical impact* of a single node on linear or linearized dynamical models in complex networks is measurable as centrality and can be predicted relatively accurately by the left eigenvector corresponding to the largest non-zero eigenvalue of a characteristic matrix of the system considering both topological and dynamical features of the network [83]. Here we try to find suitable centrality measures regarding the nodes' *dynamical impact* in Boolean networks.

4.2 Dynamical Impact

So far we have considered the average effect of a flip perturbation at the input i of a Boolean function f_j on the output (Equation 2.3). Now we ask about the long-term behaviour of the whole system after a perturbation. We define

$$H_i(t) = \{x \in \{0,1\}^N : f^t(x) \neq f^t(x^{\updownarrow i})\} \tag{4.1}$$

as the set of initial conditions such that a perturbation at node i spreads at least until time t. Then the fraction of such combinations

$$h_i(t) = \frac{|H_i(t)|}{2^N} \tag{4.2}$$

out of all possible ones is the probability that the damage spreads for at least t steps after perturbing node i. We call $h_i(t)$ the *dynamical impact* of node i for t steps (see appendix A.4). Figure 4.1 shows that dynamical impact strongly varies across nodes of a given random network. The maximum value is typically more than an order of magnitude larger than the average in networks of size $N = 500$ with $K = 2$ at critical sensitivity $\langle s \rangle = 1$.

Let us find an analytic approximation for $h_i(t)$ at long times t. By $p_i(t)$ we denote the probability that node i carries a damage at time t, i.e. the probability that $[f^t(x)]_i \neq [f^t(x^{\updownarrow i})]_i$. After the perturbation has spread for at least one time step, the damages and also the unperturbed states are correlated across nodes in general. Then the single-node probabilities $p_i(t)$ are insufficient for an exact description of the spreading probabilities. Here we make an approximation by neglecting the correlations. Then the damage probabilities follow the

[2] **Degree** is the number of neighbors [77]. Here a node has a higher rank when it has more neighbors. **K-core** is the largest sub-network of the main network where all the nodes have greater degree than K [78]. Being in higher core means having larger centrality. **K-shell** is obtained by omitting the (K-1)-core from the K-core [79]. Nodes belong to the higher shell are more important. **Betweenness** is the number of shortest paths from all vertices to all others that pass through that node [80]. So, to have a large betweenness centrality, the node must be between many of the nodes via their geodesics. **Closeness** is the mean distance from a node to the others. The closer a node is the more centrality it has. **Eigenvector centrality** is eigenvector of adjacency matrix corresponding to the largest eigenvalue [81]. A node is more influential in the system if it has more connections to other influential nodes. Read Ref. [82] for technical discussions.

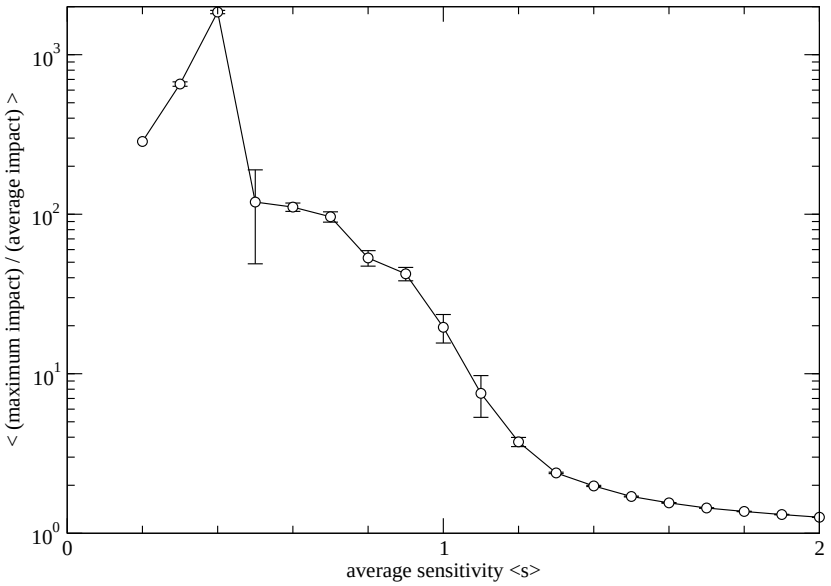


Figure 4.1: Variation of dynamical impact across nodes in random Boolean networks. For all networks, the ratio $r = [N \max_i h_i(t)]/\sum_{i=1}^{N} h_i(t)$ between maximum and average impact is calculated. Each data point is the average of r over 100 networks of size $N = 500$ with connectivity parameter $K = 2$ at the given average sensitivity. Error bars indicate standard deviation divided by 10. Dynamical impact strongly varies across nodes of a given network. For instance, the ratio between the largest and the average impact is 20 ± 4 at $t = 100$ on random Boolean networks (500,2) with $\langle s \rangle = 1$.

equation

$$p_j(t) \propto \sum_{i=1}^{N} \alpha_{ij} p_i(t-1) \ . \tag{4.3}$$

where α_{ij} is activity (Eq. 2.3). This equation is exact if the network, seen downstream from the initially perturbed node, is a directed tree. Then at most one term in the summation is non-zero. Otherwise Eq. 4.3 serves as an approximation assuming a roughly linear accumulation of the damage.

Figure 4.2 provides an illustration. In a more compact notation, Eq. 4.3 reads

$$p(t) = \aleph^{\mathrm{T}} p(t-1) \tag{4.4}$$

using the transpose of the activity matrix $\aleph = (\alpha_{ij})_{ij}$. Iteration from the initial condition yields

$$p(t) = (\aleph^{\mathrm{T}})^t p(0) \ . \tag{4.5}$$

In the limit of large t, the projections on the (left and right) eigenspaces of the leading eigenvector of $\aleph$ dominate the behaviour of p. Assuming that $\aleph$ is irreducible, non-negativity ensures that these eigenspaces are one-dimensional by the Perron-Frobenius theorem. Then

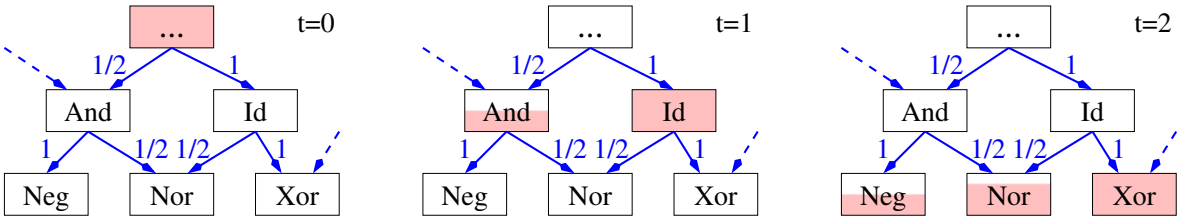


Figure 4.2: Probabilistic description of damage spreading in a Boolean network. The estimated damage probability $p_i(t)$ for a node i at time t is indicated by the height of the shaded area. At time $t = 0$, the upper node is perturbed, thus having a damage probability 1. Neglecting correlations, the probability that a damage spreads from a node i to a node j is the activity α_{ij} as a label on each connection $i \to j$. Note that the case of more than one perturbed input, such as for the node with the Nor-function, is not captured by the activities. In the analytic treatment, we assume linear superposition of damage probabilities. The node performing Nor has an estimated damage probability $(1/2)(1/2)+1(1/2) = 3/4$ at time $t = 2$.

Table 4.1: Centrality measures considered as predictors for the dynamical impact $h_i(t)$.

↓ range ↓	adjacency matrix A	activity matrix $\aleph$
local	out-degree (d_i)	strength (σ_i)
global	eigenvector (e_i)	eigenvector (ϵ_i)

we find unique normalized right and left principal eigenvectors ϵ' and ϵ of $\aleph$ with non-negative entries. In this approximation by the dominant eigenspaces, the evolution of p reads

$$p(t) = \lambda^t(\epsilon' \otimes \epsilon)p(0) \tag{4.6}$$

with the dyadic product of ϵ and ϵ' and the largest eigenvalue λ. According to Equation 4.6, the projection of the initial damage probability $p(0)$ on the eigenvector ϵ is what determines the expected damage at long time t. In other words, ϵ_i is indicative of the long-term damage expected from a perturbation at node i in the linearized treatment with suppression of correlations.

To which extent does this asymptotically expected damage amplitude ϵ_i inform us about the probability $h_i(t)$ that the perturbation spreads for a long time t? In the following sections we investigate this question by simulations. Often the network structure is known but information on the Boolean functions is lacking. Taking all non-zero activities as having value 1 turns the activity matrix into the adjacency matrix. Therefore we also consider the predictive power of the leading left eigenvector $e = (e_1, e_2, \ldots, e_N)$ of the adjacency matrix. In situations without global knowledge on the system, we may want to compare dynamical impacts of a few nodes, for which the network neighbourhood is known. Then we can resort to the strength σ_i or the out-degree d_i as centralities of node i based on local information. Table 4.1 summarizes the four centralities under consideration.

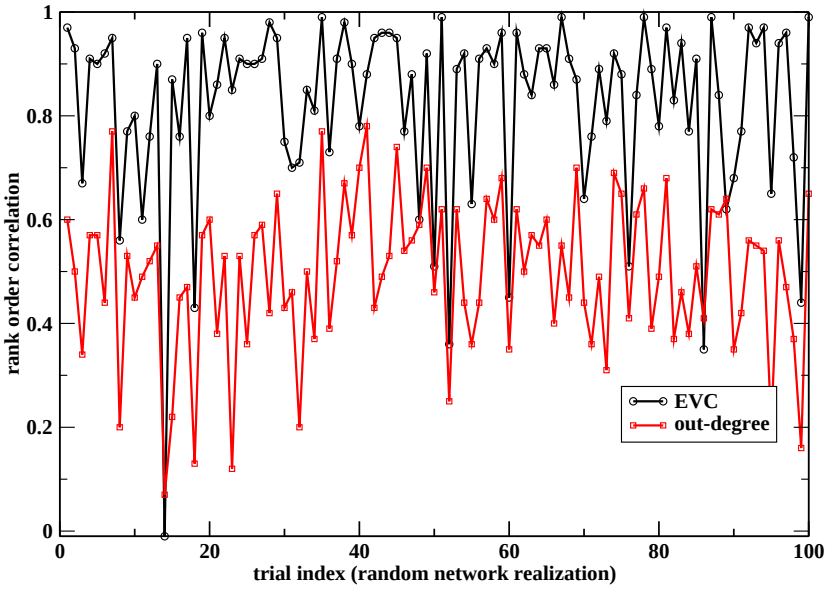


Figure 4.3: Rank order correlation between the eigenvector of adjacency matrix (black $\diamond$) and out-degree (red $\square$) measures and dynamical impact $h(t)$ of a long-term synchronous updating, for 100 independent realizations of random Boolean networks (50,2) with given sensititivity $\langle s \rangle = 1$.

4.3 Results for Random Boolean Networks

Generating 100 random Boolean networks in the critical regime ($\langle s \rangle = 1$)[3], Fig. 4.3 shows that eigenvector e is almost always a better predictor than out-degree vector d.

But now Let us investigate the dynamical impact of nodes and its prediction by centrality measures (cf. Table 4.1) on random Boolean networks with $N = 500$ nodes and connectivity parameter $K = 2$ with more details. As shown in Figure 4.4 (a), the long-term impact of perturbations is best predicted by the leading eigenvector ϵ of the activity matrix in the whole range of sensitivity. Prediction by the leading eigenvector e of the adjacency matrix is inferior to that by ϵ in the supercritical regime $\langle s \rangle > 1$. When reaching $\langle s \rangle = K = 2$, predictive powers become equal again, because all Boolean functions are exclusive-or or its negation. Then all network connections have activity value 1 and adjacency and activity matrices are the same. The superiority of the eigenvector ϵ as a predictor is in agreement with the analytic arguments given in the previous section. Slightly above the critical sensitivity value 1, predictive power shows a peak for all centrality measures considered. Further analyses of the dynamics are necessary to understand the variation of the predictive power with average sensitivity, especially the minimum of $\mathcal{P}_\epsilon$ at $\langle s \rangle \approx 0.7$.

Figure 4.4(b) displays predictive power at short times, here $t = 1$. As expected, strength σ is the best predictor in this case. Predictions by the out-degree vector d perform second best

[3]See section 4.8 for details.

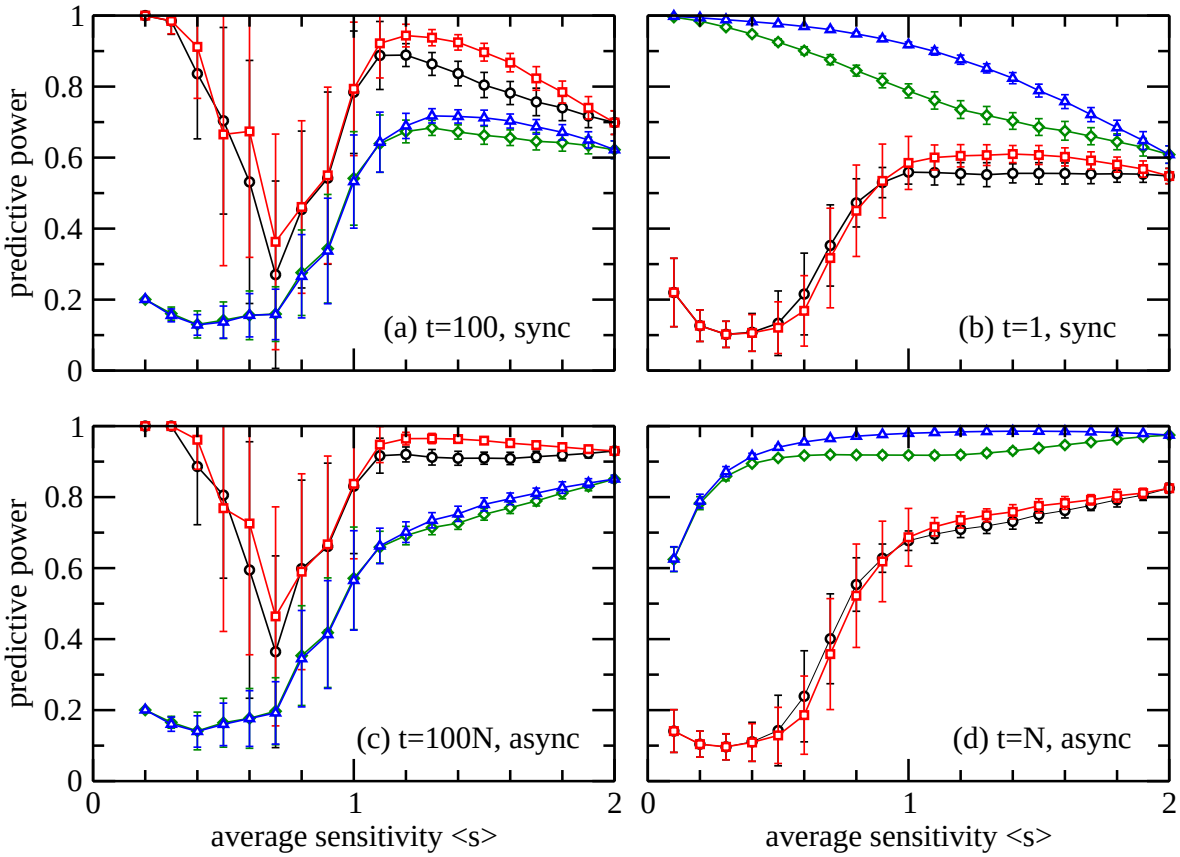


Figure 4.4: Quality of prediction of dynamical impact in random Boolean networks at varying average sensitivity $\langle s \rangle$. Symbols distinguish the centrality measures out-degree d (green $\diamond$), strength σ (blue $\triangle$), and the principal eigenvectors ϵ and e of the activity matrix (red $\square$) and the adjacency matrix (black $\circ$). The four panels present combinations of long- or short-term prediction with deterministic synchronous or stochastic asynchronous update. Each data point gives the rank order correlation (cf. section 4.8) with dynamical impact $h(t)$, averaged over 100 independent realizations of random Boolean network (500,2) with given sensititivity $\langle s \rangle$. The error bars indicate the standard deviation over realizations.

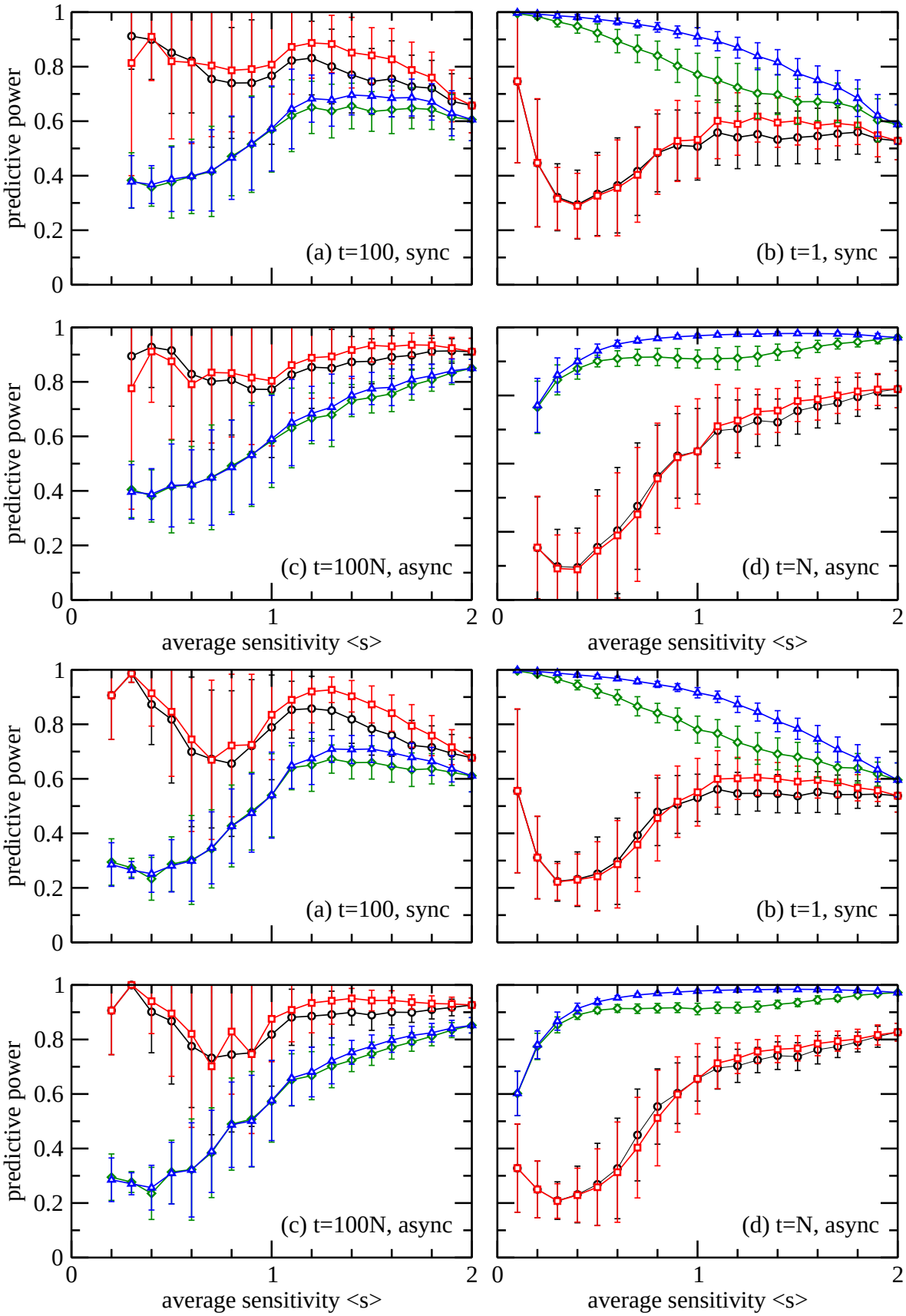


Figure 4.5: Predictive power of centrality measures for random Boolean network ensembles averaged over 100 realizations with two system sizes: $N = 50$ (four up charts) and $N = 100$ (four down charts). Symbols are same as Fig. 4.4.

but significantly worse than those by strength σ.

The results in the upper panels of Fig. 4.4 are obtained under synchronous update of the whole system, as defined by Eq. 2.1. In order to check the robustness of the results, we repeat the simulations under stochastic *asynchronous* update according to Equation 4.8. The results, shown in panels (c) and (d) of Fig. 4.4, are qualitatively similar to those obtained under synchronous update. In the super-critical regime, however, the predictive power of all four centrality measures is increased when the updating is asynchronous instead of synchronous. Thus damage spreading is easier to predict under asynchronous update, at least with the four centrality measures studied here. This effect must be rooted in the interplay between the update order and the network structure. For instance, the damage definitely heals when the perturbed node receives the first update before all its predecessors. The frequency of this happening decreases with the out-degree d_i and incurs an additional dependence of dynamical impact on the centrality measure d.

Simulations at different network sizes ($N = 50$, $N = 100$, Fig. 4.5) yield similar results for all four combinations of long- or short-term spreading and synchronous or asynchronous updates. The predictive power of all four centrality measures remains constant or increases with system size.

4.4 Switching Between Attractors

As the long-term behaviour of Boolean dynamics is determined by attractors, it is natural to ask if a perturbation in the initial condition will cause the system to arrive at a different attractor. For this investigation, we define the attractor impact h'_i of node i as the fraction of initial conditions where a perturbation at node i changes the attractor eventually reached. At difference with dynamical impact $h_i(t)$, attractor impact h'_i does not set an explicit time t after which to determine the spreading or healing of the perturbation. On the other hand, h'_i does count the perturbation as healed whenever the perturbed and unperturbed dynamics eventually become equal up to a time lag.

Figure 4.6 shows the predictive power of the centrality measures for attractor impact of nodes. For averaged values, the performance comparison yields $\mathcal{P}_\epsilon > \mathcal{P}_e > \mathcal{P}_\sigma > \mathcal{P}_d$, being the same as for predicting long-term dynamical impact (cf. Figure 4.4 (a,c)). Due to fluctuations around the averages, this ordering does not hold for each single realization. At each considered value of the average sensitivity, a fraction at least 3/4 of the realizations has ϵ as the best predictor. Sub-critical networks, $\langle s \rangle < 1$, are disregarded here because most realizations do not have more than one attractor. For $\langle s \rangle > 1.9$, attractor search exceeds available computer time. Also Fig. 4.7 indicates how much predictive powers change by this alternative of dynamical impact $h_i(t)$.

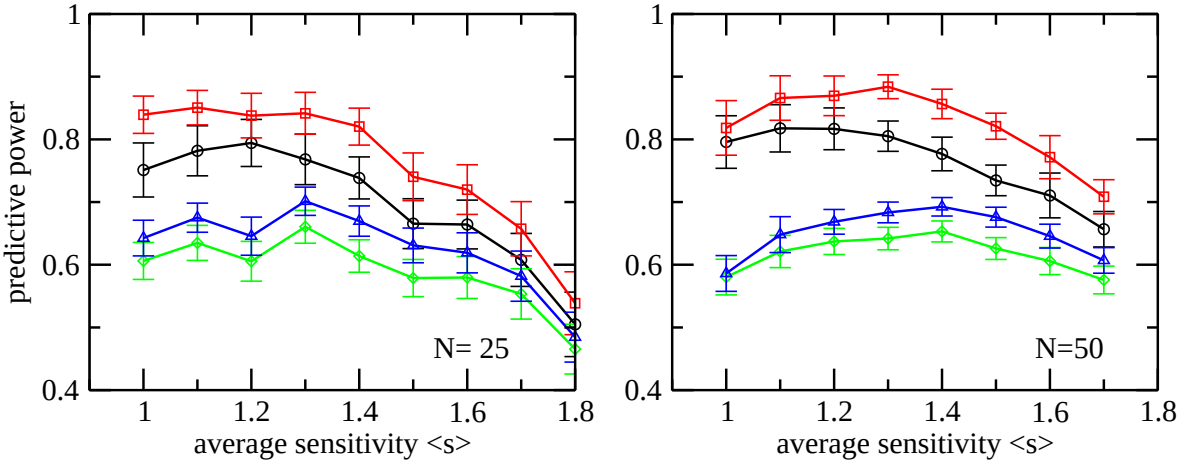


Figure 4.6: Power of centrality measures for predicting if a perturbation changes the attractor reached. Symbols for the centrality measures are the same as in Fig. 4.4. Each data point is an average over 100 random Boolean networks with $N = 25$ (left), $N = 50$ (right) nodes and connectivity parameter $K = 2$. Error bars indicate the standard deviation over the random network ensemble. Error bars are scaled down by the factor 0.2 to avoid overlapping.

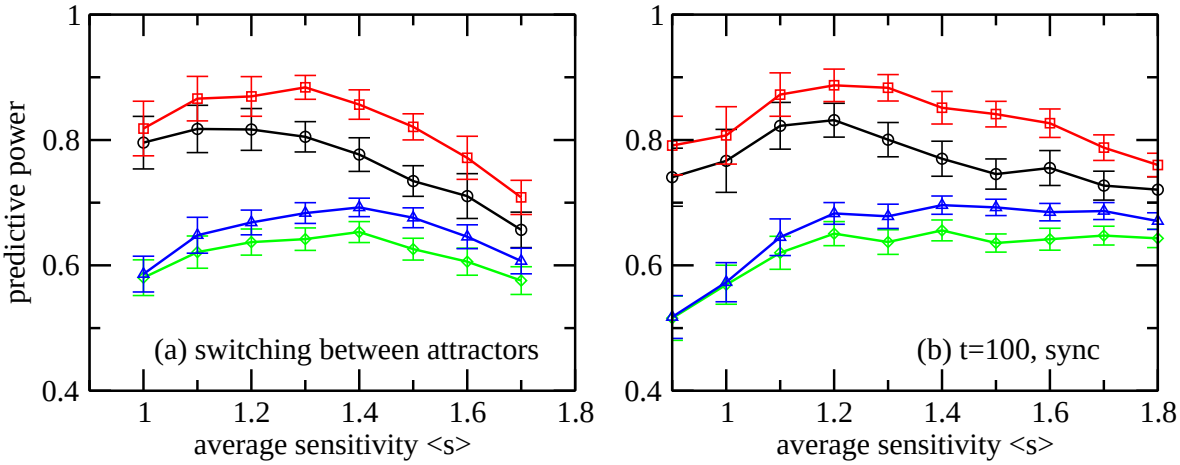


Figure 4.7: A comparison of predictive powers when dynamical impact was based on attractors switching (left) and long-term synchronous updating (right). Points are averaged over 100 random Boolean networks (50,2). Error bars indicate the standard deviation over the random network ensemble. They are scaled down by the factor 0.2 to avoid overlapping. Symbols are similar to Fig. 4.4.

Table 4.2: Predictive power of centrality measures for the mammalian cell cycle, yeast cell cycle and Th network. Rows distinguish update modes in long- or short-term dynamics. The bold number are the maximum in each line.

		mammalian cell cycle			
		$\mathcal{P}_\epsilon$	$\mathcal{P}_e$	$\mathcal{P}_\sigma$	$\mathcal{P}_d$
synchronous	$t = 1$	0.778	0.328	**0.948**	0.298
	$t = 100$	**0.842**	0.515	0.744	0.456
asynchronous	$t = N$	**0.721**	0.406	0.683	0.344
	$t = 100N$	0.406	-0.174	**0.409**	-0.0599

		yeast cell cycle			
		$\mathcal{P}_\epsilon$	$\mathcal{P}_e$	$\mathcal{P}_\sigma$	$\mathcal{P}_d$
synchronous	$t = 1$	0.636	0.327	**0.938**	0.676
	$t = 100$	0.409	0.091	**0.613**	0.364
asynchronous	$t = N$	0.727	0.455	**0.741**	0.604
	$t = 100N$	0.445	0.100	**0.540**	0.297

		Th network			
		$\mathcal{P}_\epsilon$	$\mathcal{P}_e$	$\mathcal{P}_\sigma$	$\mathcal{P}_d$
synchronous	$t = 1$	0.212	-0.208	**0.854**	0.353
	$t = 100$	0.589	0.317	0.767	**0.777**
asynchronous	$t = N$	0.336	-0.0612	**0.963**	0.659
	$t = 100N$	0.592	0.340	0.713	**0.770**

4.5 Dynamical Impact in Real Boolean Networks

Let us now test the performance of predictors on non-random networks introduced in Sec. 2.7. Table 4.2 indicates the predictive power of centrality degree measures for the dynamical impact of nodes in the first three mentioned Boolean dynamics. For these networks, node's strength σ_i predicts the best short-term perturbation spreading. As dynamics end up at a limit cycle relatively in a short time, eigenvector centrality measures, ϵ_i and e_i, are not as good predictors as we expected in long-term.

In table 4.3, we summarize the predictive power of centrality measures for the dynamical impact of nodes in the fibroblast network. Also for this network, the leading eigenvector ϵ of the activity matrix is the best predictor of a node's ability to cause long-term spreading of a perturbation. short-term spreading is best predicted by a node's strength σ_i. Table 4.4 shows the five nodes with the largest dynamical impact and their ranks with respect to the centrality measures. Prediction of these ranks by the centrality measures is not perfect. However, the leading eigenvector ϵ of the activity matrix correctly identifies four out of the five nodes with the largest impact.

Table 4.4 and the lower part of Table 4.3 are obtained for the fibroblast network after removal of the nine input nodes. These nodes indefinitely sustain their state. Therefore a

Table 4.3: Predictive power of centrality measures for the fibroblast signal transduction dynamics. The upper part of the table considers the original system. The lower part is for the system after removal of the nine nodes providing constant input. Each line of the table is a scenario defined by the update mode and the choice of long- or short-term dynamics. The bold number indicates the maximum in each line.

		all nodes			
		$\mathcal{P}_\epsilon$	$\mathcal{P}_e$	$\mathcal{P}_\sigma$	$\mathcal{P}_d$
synchronous	$t = 1$	0.671	0.454	**0.930**	0.455
	$t = 100$	**0.920**	0.734	0.746	0.523
asynchronous	$t = N$	0.706	0.528	**0.904**	0.564
	$t = 100N$	**0.854**	0.694	0.748	0.542
		only core nodes			
		$\mathcal{P}_\epsilon$	$\mathcal{P}_e$	$\mathcal{P}_\sigma$	$\mathcal{P}_d$
synchronous	$t = 1$	0.633	0.467	**0.946**	0.528
	$t = 100$	**0.911**	0.777	0.738	0.611
asynchronous	$t = N$	0.658	0.543	**0.919**	0.656
	$t = 100N$	**0.834**	0.731	0.741	0.631

Table 4.4: The five core nodes of the fibroblast network with the largest dynamical impact and their ranks with respect to the four centrality measures. Synchronous update is performed on the core of the network, after removal of the nine input nodes. Dynamical impact $h_i(t)$ measures spreading over $t = 100$ time steps.

node i	$h_i(100)$	$r_i(\epsilon)$	$r_i(e)$	$r_i(\sigma)$	$r_i(d)$
Src	0.7707	1	1	1	1
B-Arrestin	0.7061	4	4	9	14
GRK	0.6458	16	27	17	43
PIP2-45	0.5961	2	12	4	4
PKC	0.5910	3	13	3	5

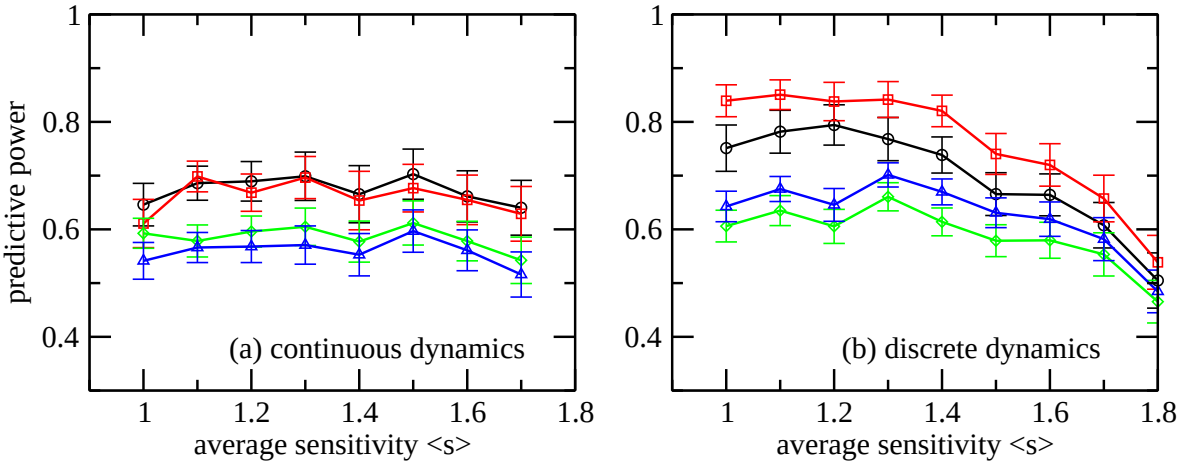


Figure 4.8: Power of ranking strategies for dynamical impact if the perturbation changes the attractor reached in a random Boolean ensemble of 100 realizations with $N = 25$ nodes, $K = 2$ by (a) continuous and (b) discrete dynamcis. Error bars indicate the standard deviation over the random network ensemble. They are scaled down by the factor 0.2 to avoid overlapping. Presented symbols are same as Fig. 4.4.

perturbation at an input node i never heals, yielding maximal dynamical impact $h_i(t) = 1$ for all times t. Reduction to the dynamical core by the removal of the input nodes allows for a less biased assessment of predictive power.

4.6 Ranking Strategies and Continuous Boolean Dynamics

Quantifying dynamical impact of nodes, we have so far considered the effect of a flip perturbation on Boolean dynamics. But we can try another alternative by considering continuous Boolean dynamics defined in Eq. 2.9 in Sec. 2.6. This alternative is small perturbations[4].

Figure 4.8 shows that ranking strategies are working pretty well here too. But weighting the adjacency matrix does not improve the predictive power, since information of the Boolean functions is somehow lost by the chosen transfer functions.

4.7 Discussion

Even in random Boolean networks, nodes exhibit significant differences in dynamical impact. As we have shown here, these differences are captured well by local and global centrality measures. From a linearization of the dynamics, these centralities arise as column sums and eigenvectors of a network matrix. Practical applications therefore benefit from efficient computation as compared to costly direct simulation of damage spreading.

[4]See more details and technical discussions about small perturbations in Sec.5.2.

An important implication for networked dynamical systems is the possibility of capturing response to perturbations based on incomplete information about the system's structure. Here, prediction of dynamical impact only uses the activity matrix while being ignorant of the actual rule tables. Hence no distinction between positive and negative feedback loops enters the calculation. Large systems of interest, such as the networks studied in this contribution, have a rich structure with many intertwined positive and negative feedback loops. We speculate that the competing effects of the two types of feedback balance when averaging over perturbations, thus allowing to predict dynamical impact without explicit knowledge of feedback types. Future investigation of this idea, possibly in smaller systems with fewer feedback loops, is needed to establish a criterion for the predictability of dynamical impact based on partial information.

The scenario of predicting node impact based on partial knowledge is particularly relevant for biological systems where not all interactions are known in full detail. A large number of measures has been suggested for identifying the dynamical centers of biological systems based on the underlying network structure alone [84, 78, 83]. Most of these approaches provide only an intuitive understanding of the assumed correlation between the centrality in the network and impact on the dynamics. The present framework, beside its accuracy and computational efficiency, is based on a verifiable description of the system's response to perturbations. In particular, it allows us to distinguish between short and long-term effects. The result is a detailed set of predictions testable in experiments.

4.8 Methods

Random Boolean networks. A random instance of a Boolean network with N nodes, connectivity parameter K and expected average sensitivity $\langle s \rangle$ is generated as follows. Each node i is assigned a Boolean function f_i, drawn from the distribution

$$\pi(f) \propto \exp[\lambda s(f)] \; . \tag{4.7}$$

The distribution π is normalized and supported by the set of 2^K Boolean functions with at most K inputs. Here λ is chosen such that the expectation value of $s(f)$ under the distribution π is equal to the average sensitivity $\langle s \rangle$ [31]. Then π is the unique distribution maximizing entropy with the given $\langle s \rangle$ (see appendix A.1). For each input, on which f_i actually depends, a link (j, i) is established with the source node j drawn uniformly at random. When this would lead to a duplicate or self-coupling, j is discarded and redrawn.

Both for the random and the empirical Boolean networks, we estimate dynamical impact of a node i by 10^4 runs of the dynamics. For each of these, a state $x(0) \in \{0, 1\}$ is drawn uniformly. Then two replica of the system are initialized with $x(0)$ and at $(x(0))^{\updownarrow i}$. The fraction of runs where the replica are in different states at time t is taken as approximation

of $h_i(t)$ (see appendix A.4). When $h_i(t)$ is the same for all nodes i or the largest eigenvalue of the network's activity matrix is degenerate (see appendix A.5), the network is discarded and a new independent realization is drawn. Discarding of network happens mostly at small $\langle s \rangle$. It does not occur in any of the trials with $\langle s \rangle \geq 1.2$.

The dynamics of Equation 2.1 is deterministic with synchronous update. Alternatively, we consider stochastic asynchronous update as follows. At each time step t, a node $u(t)$ is drawn uniformly at random and the nodes take states

$$x_i(t+1) = \begin{cases} f_i(x(t)) & \text{if } i = u \\ x_i(t) & \text{otherwise} \end{cases} \tag{4.8}$$

in the subsequent time step. The same random sequence $u(t)$ of updated nodes is used for the perturbed and the unperturbed replica of the system.

Predictive power. We quantify the predictive power $\mathcal{P}_y$ of a centrality measure $y \in \{d, e, \sigma, \epsilon\}$ as the rank order correlation with dynamical impact.

$$\mathcal{P}_y = \mathrm{corr}(r(h), r(y)) \tag{4.9}$$

with the usual Pearson correlation coefficient corr. For a general vector $v = (v_1, v_2, \ldots, v_N)$, the rank vector $r(v)$ has entries

$$r_i(v) = 1 + |\{j \neq i | v_j > v_i\}| + \frac{1}{2}|\{j \neq i | v_j = v_i\}| \ . \tag{4.10}$$

See appendix A.6 and A.4.

CHAPTER 5

Stability of Boolean Dynamics

> ""Stability," said the Controller, "stability. No civilization without social stability. No social stability without individual stability.""
>
> *Chapter 3, Brave New World*
> Aldous Leonard Huxley (1894-1963)

Continuing our discussions about stability in different models of gene regulatory dynamics in chapter 3, let us now focus on our main case, Boolean networks. Studying the response of Boolean dynamics to a perturbation is a key concept to understand the dynamics better. Numerous works have recently focused on assessing the reactions of Boolean networks to a failure. The following questions are the main concerns: how or how far and long can a damage spread in the networks or the state space of the system? Can dynamics overcome the imposed failure in the short or in the long run? In order to answer these questions, let us have a quick overview of the literature on Boolean networks to see first of all what can be considered as a perturbation and secondly what can be used as the metric to measure the smallness or largeness of the applied perturbations and the outcoming variations in dynamics at a later time. Finally we try to answer the question of whether or not the well-known stability criterion, flip perturbation, keeps the consistency when changing mathematical frameworks from continuous delay differential equations to discrete Boolean dynamics discussed in section 2.6[1].

[1]This chapter is adopted from [31].

5.1 Introduction

For testing structural robustness as well as dynamical resilience, one can perturb the system by manipulating any mathematical element of the framework defined in chapter 2, such as nodes, links, Boolean functions, updating schemes and states of nodes. Let us present some attempts to study the stability behaviour of Boolean dynamics in terms of different mathematical bases.

Nodes: The first available option to manipulate a Boolean dynamics is to kill a node (or a set of nodes), which means removing the node's interactions with the others in the network and it is known as node-knockout. The network is robust against this damage whenever it can still perform the original dynamics. For instance, Boldhaus et. al. have studied the effect of knock-outs in threshold Boolean networks to compare a null model resilience with the yeast wild-type network [69].

Links: Another option is to make a link-knockout, which means killing only a link or a set of links. Note that if the deleted set includes all interactions of one node, then a node-knockout has happened.

Boolean Functions: As discussed in section 2.5, studying the stability behaviour of dynamics is another method for understanding the dynamical phase of the system. Some studies have been devoted increasing our knowledge about the critical behaviour of Boolean networks, with different ensembles of functions [44, 45]. As an intersting example, Moreira and his colleagues have analytically and numerically shown the role of canalizing functions on critical transition of Boolean dynamics [85].

Updating Schemes: Gershenson has investigated the effect of six types of updating schemes, synchronous or asynchronous, deterministic or non-deterministic, on the resilience of loosely defined attractors against damage. He concludes that critical stability on the border of order and disorder phases does not depend on the updating schemes [36].

Flip Perturbation: In the state-*discrete* Boolean dynamics, *large* perturbations are normally implemented as a *flip*, where the state of a single Boolean variable is inverted. Then the evolution of the damage is tracked. The damage is the difference between the state of the perturbed and the unperturbed system. The return map of the expected size of the damage is known as Derrida plot [50]. Numerous studies have elucidated the effect of flip perturbations on regulatory dynamics with Boolean states [66, 51, 86, 87, 12, 53]. Most of the investigations of phase transition of Boolean dynamics discussed in section 2.5 were based on this criteria in the limit of large N.

Time Lags: One interesting approach to building a bridge between continuous and discrete stability analyses considers time as a continuous parameter in the system. This allows the modellers to test reliability of the dynamics with infinitesimally small perturbations. Making time-lags to fluctuate the switching times of the dynamics, one can test the reproducibility of the dynamics, i.e. whether it can synchronize again in its own fashion with or without phase lag or rather desynchronizing. The former is a reliable dynamics and the latter is an unreliable one [73].

Conflicting some recent numerical results [69, 22] with considering the size of the basin of attractions as a measure of robustness [25] on the one hand, and the strong correlation of the stability behaviour with the phases classifications of the system, which can represent the evolvability of the dynamics too [48], on the other hand, brings a question to mind that where, which criteria can present stability or instability of the dynamics best fit. So let us now proceed, in the next section, to our final important regarding comparison of different Boolean stability criteria.

5.2 Small Perturbations

When asking if a gene-regulatory system reproduces a prescribed trajectory despite noise, large perturbations are to be considered in the case of low copy numbers of regulatory molecules and bursty stochastic response [72]. Small perturbations, however, are more appropriate when modeling systems with large copy numbers and an integrative response to filter out bursts (see e.g. [76]).

Here we find that the clear distinction between the two types of perturbations is crucial. In a continuous system, stability or instability under small perturbations is not indicative of the effect of flip perturbations. Likewise, probing a Boolean system with flip perturbations does not necessarily provide information about the stability of the continuous counterpart under small perturbations.

In other words, we want to show that the dynamics of large random networks of switch-like elements typically recovers from small perturbations of the state vector. Healing is observed naturally at low sensitivity. i.e. a weak dependence of the logic functions on their inputs. Surprisingly, however, also large sensitivities of functions lead to a system behaviour that is insensitive to small perturbations in the initial condition. Inherent instability is observed only in an intermediate sensitivity regime that shrinks as systems become larger.

This finding may at first sight appear to be incompatible with the established stability diagram of random Boolean networks, being the time- and state-discrete representations of the aforementioned systems of switch-like elements. Random Boolean networks display a transition from healing to non-healing (damage spreading) behaviour at a critical average sensitivity of 1, when subject to *flip* perturbations. These particular perturbations tend not

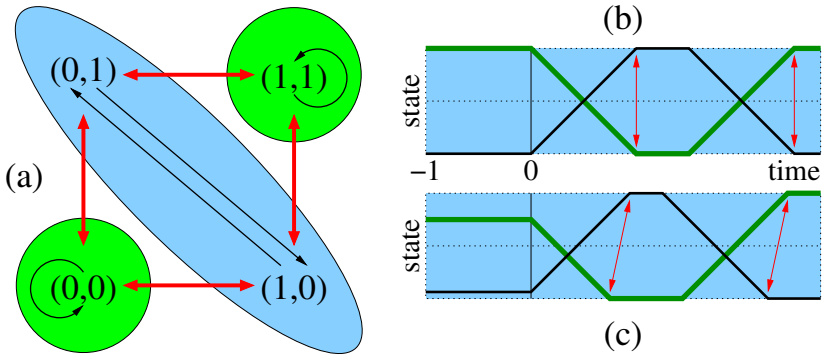


Figure 5.1: Dynamics of two mutually activating nodes. (a) State space of the Boolean system described by Eq.5.1. Thin arrows indicate the mapping f of states by the dynamics, thick bidirectional arrows stand for flip perturbations. Indicated by shaded areas, the system has three dynamical modes (attractors): two fixed points $(0,0)$ and $(1,1)$ and a cycle of length 2 involving the states $(0,1)$ and $(1,0)$. (b) Time evolution of the corresponding continuous system in Equation 2.9 with initial condition $x_1(0) = 1$ (thick curve) and $x_2(0) = 0$ (thin curve). The two nodes switch in a synchronous mode as indicated by vertical double arrows akin to the Boolean state sequence $(0,1),(1,0),(0,1),\ldots$. (c) Time evolution from perturbed initial condition, $x_1(0) < 1$, $x_2(0) > 0$. The perturbation translates into a phase lag in switching that does not heal out.

to heal in random networks of highly sensitive functions. The tacit assumption that flip perturbations are sufficient to probe the stability of the dynamics lead to the hypothesis that networks of regulatory switches position themselves at this transition [48], known as the *edge of chaos* [88]. This would cause some but not all flip perturbations to spread and therefore allow for complex information processing without rendering the system unreliable under noise.

5.3 Fixed points and bistable circuits

Let us first consider a fixed point as the simplest dynamical behaviour. A fixed point of the continuous dynamics is a state vector y^* such that constant $y(t) = y^*$ is a solution of Eq. 2.9. This in turn means that the time derivative vanishes at all times, equivalent to $y^* = f(y^*)$. The fixed points of the continuous dynamics are exactly the fixed points of the discrete map f. A small perturbation to a fixed point y^* always heals, because values after applying the threshold Θ remain unchanged, $\tilde{f}(y'(t)) = y^*$ for all $t \in I$. All fixed points are stable under small perturbations. However, a flip perturbation to a fixed point does not always heal. The *bistable switch* is an example. Consider a two-dimensional map f with $f(x_1, x_2) = (x_2, x_1)$. It gives rise to the dynamics

$$x_1(t+1) = x_2(t) \qquad x_2(t+1) = x_1(t) \tag{5.1}$$

with fixed points $(0,0)$ and $(1,1)$. After perturbing a fixed point by flipping one node's

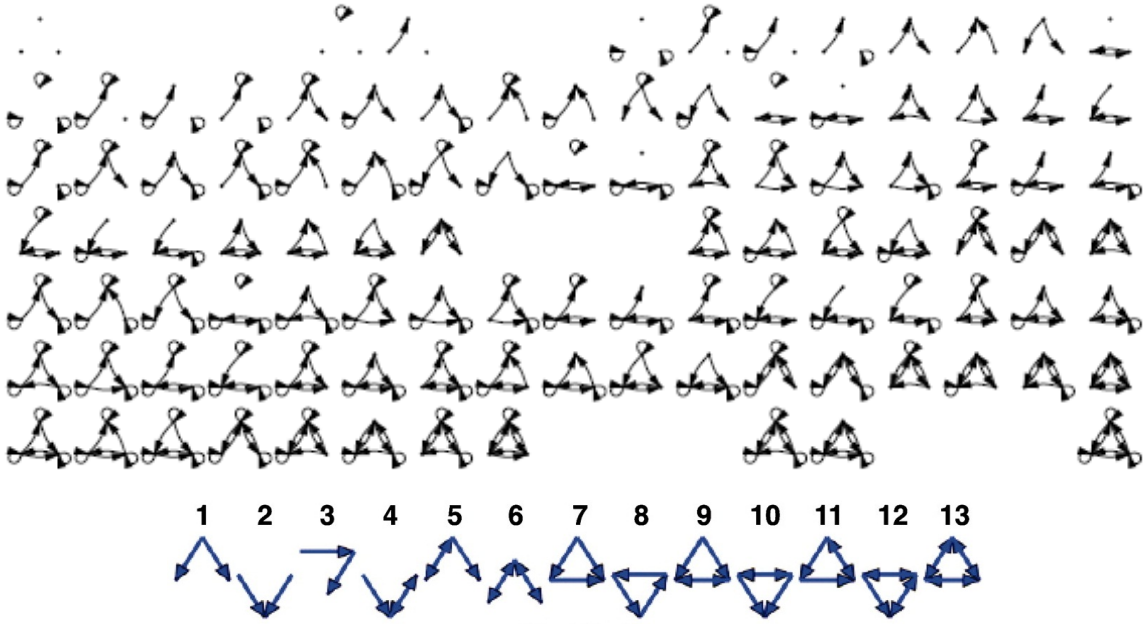


Figure 5.2: There are 104 distinct directed graphs of 3 nodes shown in black. Restricting ourselves to non-self-interactive connected graphs, we will only have 13 choices shown in blue. Upper figure is taken from an unpublished paper of Thomas Fink and his colleagues and the lower from [73]

state, the system does not return to the fixed point. It remains in the set of the state vectors $(0, 1)$ and $(1, 0)$ constituting a limit cycle (cf. Figure 5.1(a)). The stability of the fixed points is not obtained when probing the dynamics with flip perturbations. The bistable switch constitutes a first simple example of systems with different stability properties under flip and small perturbations.

In the continuous counterpart of the alternating Boolean state $(0, 1)$ and $(1, 0)$, small perturbations do not heal (see Figure 5.1(b,c)). The effect of a small perturbation is to induce a phase lag in the oscillation, being discussed in earlier works [73, 49, 74, 63].

5.4 Stability of $(3, 2)$ Boolean Networks

Some researches have focused on probing the dynamics and stability of motifs, the building blocks of networks [65, 89]. Klemm and Bornholdt have realized that the most reliable 3-node networks among the thirteen shown in Fig. 5.2 are abundant in the empirical data as well; And this can be a sign of evolutionary superiority of the topology [73, 90]. Let us now study more complicated cases than fixed points, i.e. Boolean attractors of a simple motif, to see how the dynamics reacts to small and large perturbations. Here we consider only one topology, the number 13 in Fig. 5.2, with 10 different Boolean functions tabled in 5.1. We

Table 5.1: Numbering of Boolean functions and corresponding parameters of h function.

n	operator	a	b_1	b_2	c
0	$x_{in1} \vee x_{in2}$	0	0	0	1
1	$\neg x_{in1} \wedge \neg x_{in2}$	1	-1	-1	1
2	$\neg x_{in1} \wedge x_{in2}$	-1	0	1	0
3	$x_{in1} \wedge \neg x_{in2}$	-1	1	0	0
4	$x_{in1} \oplus x_{in2}$	-2	1	1	0
5	$\neg x_{in1} \vee \neg x_{in2}$	-1	0	0	1
6	$x_{in1} \wedge x_{in2}$	1	0	0	0
7	$\neg(x_{in1} \oplus x_{in2})$	2	-1	-1	1
8	$\neg x_{in1} \vee x_{in2}$	1	-1	0	1
9	$x_{in1} \vee \neg x_{in2}$	-1	0	-1	1

Table 5.2: Classification of 60 stable Boolean attractors of $(3, 2)s$ due to stability of continuous counterpart dynamics.

class	#	linear transfer function	exponential transfer function
024	11	weakly stable[a]	unstable[b]
017[c]	15	weakly stable[a]	weakly stable[a]
027	7	stable	weakly stable[d]
011	27	stable	stable

[a] Ignoring the small peak (in some cases peaks) appearing in the continuous dynamics, it's stable.
[b] Dynamics damped to a fixed point before applying any perturbation.
[c] In these cases, Δ is 0. Consult the threshold method in appendix A.3 for details.
[d] Enlarging the α, small peaks disappear in the continuous dynamics. Then it shows stable behaviour.

only considered those combinations whose numbering according to 5.1 satisfies $A < B < C$, given the way node functions are chosen. Let us first employ some examples to clarify our notation. We give the Boolean network presented in Fig. 2.3 the name 116, due to the fact that the Boolean functions of nodes A, B and C taking inputs from (B, C), (A, C), (A, B) are, respectively, the logical operators $nand(1)$, $nand(1)$, $and(6)$ from table 5.1. Or as another example, $(ABC) = (024)$ means that $(B \vee C, \neg A \wedge C, A \oplus B)$ is the set of rules updating (ABC) Boolean network. Thus the corresponding continuous dynamics is characterized by the equation

$$\dot{x_i}(t) = \alpha[\Theta(h_i(t-1)) - x_i(t)] \tag{5.2}$$

with step function Θ ($x_{threshold} = \frac{1}{2}$) and $h_i(t) = ay_j(t)y_k(t) + b_1y_j(t) + b_2y_k(t) + c$; where parameters a, b_1, b_2 and c take their value from table 5.1, and node i takes inputs from nodes j and k. Transfer functions were chosen first linear and then exponential. The ensemble includes 133 attractors ignoring fixed points, while 60 of those are stable under flip perturbations. Table 5.2 denotes 4 different classes of these stable Boolean attractors according to their response to small perturbations.

5.5 Stability in random networks

We now compare the effects of the two types of perturbations on dynamics in randomly generated networks. An ensemble of random Boolean networks [21] is generated by the number of nodes N, the number of inputs K of each node, and the probability distribution of Boolean functions $\pi(f)$ in a maximum entropy ensemble $\pi_\lambda(f) \propto \exp(\lambda s(f))$ under a given average sensitivity $\langle s \rangle$ (see methods 5.7). For random Boolean networks, where the K inputs of each node are drawn randomly and independently from the set of N nodes, the average sensitivity $\langle s \rangle$ is the crucial parameter determining the system's response to flip perturbations [32]. In the limit $N \to \infty$, these perturbations heal in ensembles with $\langle s \rangle < 1$; they spread when $\langle s \rangle > 1$. This change of behaviour in dependence of $\langle s \rangle$ is reproduced in Figure 5.3 (dashed lines) for varying K and N.

As our main result, we show in Fig. 5.3 that the $\langle s \rangle$-dependence of the healing probability under flip perturbations is qualitatively different from that under small perturbations. Only in the so-called critical region of $\langle s \rangle \approx 1$, small perturbations spread. Both for $\langle s \rangle \ll 1$ and $\langle s \rangle \gg 1$, the healing probability tends towards 1. This effect is enhanced by increasing system size. In the limit of $N \to \infty$ one may expect a finite probability of non-healing only at $\langle s \rangle = 1$. Then the dynamics is almost always stable under small perturbations.

The average time t_{heal} to heal from small perturbations increases moderately with system size as shown in Figure 5.4. For average sensitivity above 1, we observe a linear increase $\langle t_{\text{heal}} \rangle \propto N$. For lower values of the average sensitivity, the increase is sublinear.

The dynamics we have studied so far is simple but not the only possibility to pass from the

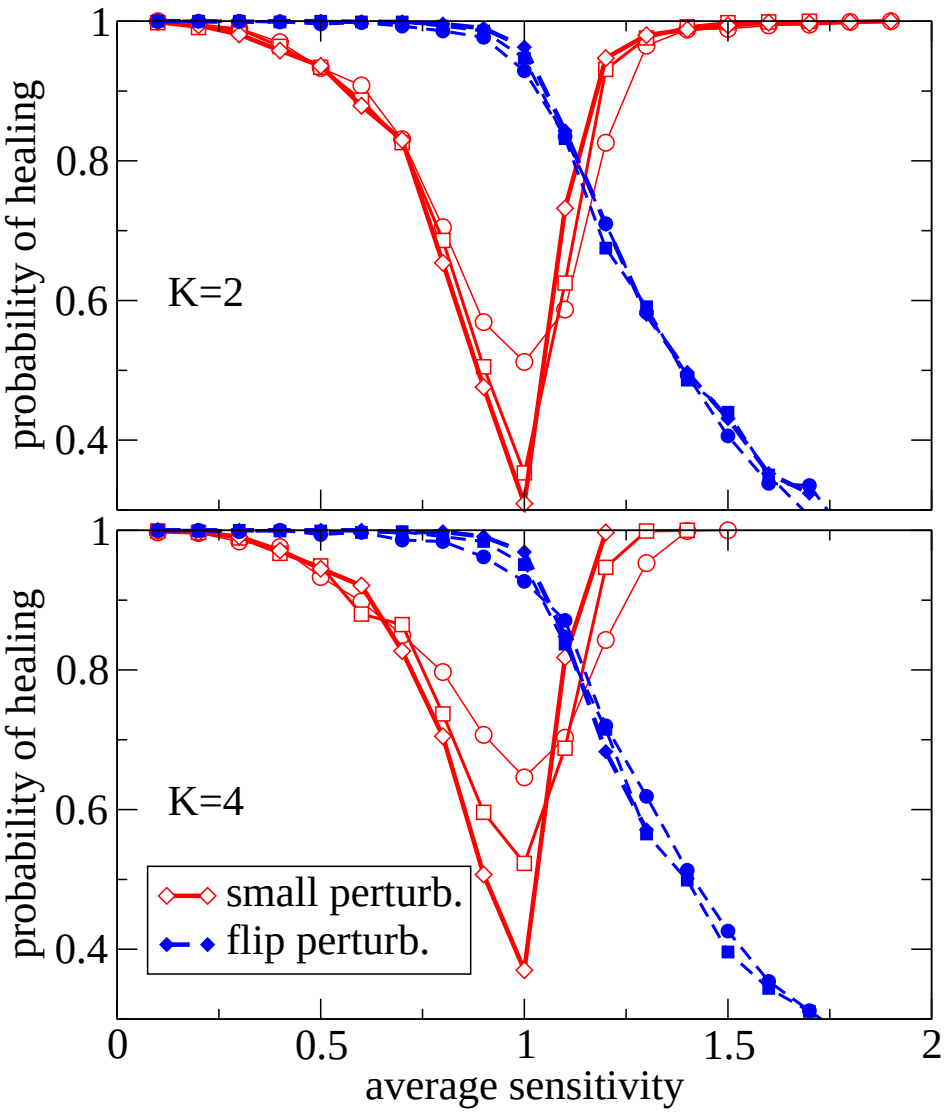


Figure 5.3: Stability of dynamics in random networks under perturbation by spin flip (dashed) and under continuous perturbation (solid lines) in random networks with $K = 2$ and $K = 4$ inputs per node. Symbols distinguish system size $N = 300$ (○), 1000 (□) and 3000 (◇). Each data point gives the relative frequency of healed out perturbations on a set of 10^4 independent random realizations of network, initial condition and perturbation. Each amplitude ϵ_i of a small perturbation is drawn independently from the uniform distribution on an interval $[0; r]$ with $0 < r < 0.5$. The results are independent of the choice of r. As a general invariance of the dynamics of Equation (2.9) with $\tilde{f} = f \circ \Theta$, the qualitative effect (healing or spreading) of a small perturbation is not altered when the amplitude vector is multiplied with a positive scalar keeping each amplitude $\epsilon_i < 0.5$. See methods 5.7 and appendix A.3 for technical details.

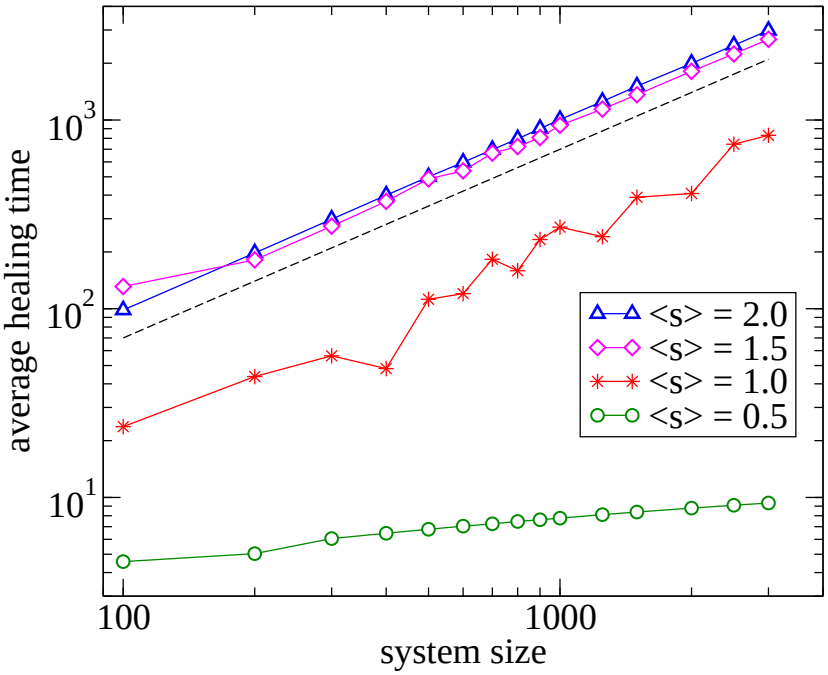


Figure 5.4: The average time to heal from a small perturbation increases linearly with the number of nodes in the system for sensitivity $\langle s \rangle \geq 1$, and sublinearly otherwise. The dashed line has slope 1 in this double-logarithmic plot. Each data point is the average over t_{heal} for the subset of healing realizations. Realizations of network, initial condition and perturbation are the same as in Figure 5.3.

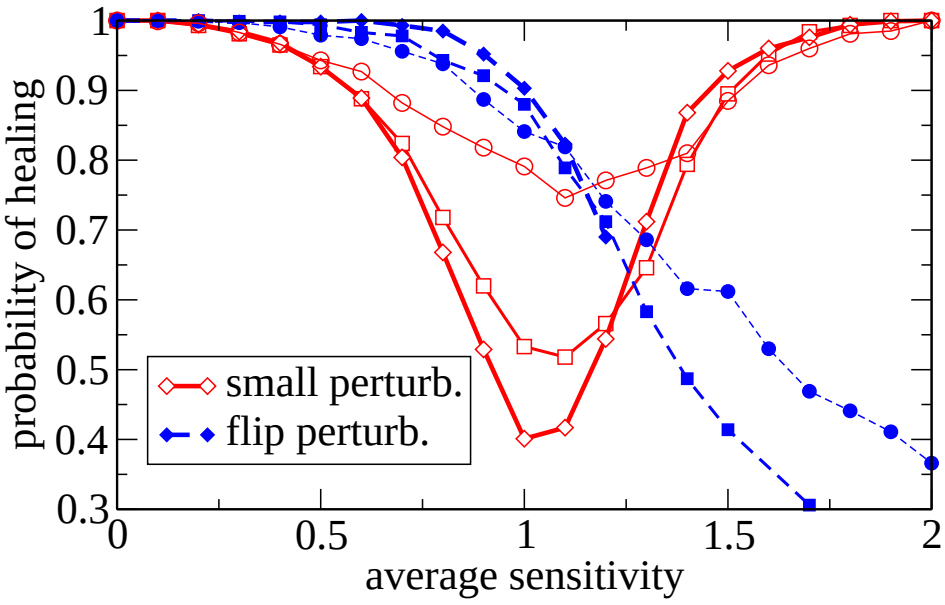


Figure 5.5: Healing probabilities remain qualitatively the same (cf. Figure 5.3) when using the alternative transfer function $\tilde{f}_i(y) = \Theta(h_i(y))$ with $h_i(y) = a y_j y_k + b_1 y_j + b_2 y_k + c$; for node i taking inputs from nodes j and k. The parameters a, b_1, b_2, c are chosen such that $h_i(y) = f_i(y)$ for $y_j, y_k \in \{0, 1\}$. If, for instance, f_i is an AND then $a = 1$ and $b_1 = b_2 = c = 0$ so $\tilde{f}_i(y) = 1$ if and only if the product of inputs $y_j y_k \geq 1/2$. Each data point is the healing fraction of 1000 realizations of given average sensitivity and system size $N = 30$ ($\circ$), 100 ($\square$) and 300 ($\diamond$). The perturbation amplitude ϵ_i is drawn from the uniform distribution on $[0; 0.01]$ independently for each node i. See methods 5.7 and appendix A.3 for technical details.

Boolean map to a continuous flow. In order to check to what extent our results depend on this choice we repeat simulations for $K = 2$ with an alternative function $\tilde{f}$ (cf. Equation 2.9) now taking into account cooperative effects between inputs. Figure 5.5 shows that the same qualitative result obtains under this choice (see figure caption for details).

5.6 Discussion

In summary, we have shown that the dynamics of large random networks of switch-like elements typically recovers from small perturbations of the state vector. Healing is observed naturally at low sensitivity. However, also large sensitivities of the nodes' functions render the long-term behaviour of the whole system insensitive to small perturbations. Instability is observed only in an intermediate sensitivity regime that shrinks as systems become larger.

The behaviour under small perturbations is essentially different from the established stability diagram for random Boolean networks discussed in section 2.5. Under *flip* perturbations, random Boolean networks display a transition from healing to non-healing (damage spreading) behaviour at average sensitivity 1. It has been suggested that networks of regulatory switches position themselves at this transition [48], known as the edge of chaos [88]. Then some but not all flip perturbations spread and therefore allow for complex information processing without rendering the system unreliable under noise.

According to our findings, a complementary scenario is worth discussing. The apparent conflict between responsiveness to external input signals and resilience to intrinsic noise dissolves when these influences act as perturbations at separate scales: noise corresponds to small perturbations whilst input signals are interpreted as the flipping of a state. Under these assumptions, noise resilience and responsiveness are compatible rather than conflicting in the regime of average sensitivity above 1. Systems that combine both beneficial properties are obtained "for free" in random networks of sufficiently sensitive switching elements.

5.7 Methods

Random Boolean Networks A random Boolean network with N nodes and in-degree K is generated as follows. For each node i, draw K pairwise different nodes $s(i,1), s(i,2), \ldots, s(i,K)$ from which i receives input. A self-coupling $s(i,j) = i$ is not allowed. Then for each node i, a Boolean function $f_i : \{0,1\}^K \to \{0,1\}$ is drawn from a distribution π. This distribution π defines a statistical ensemble of Boolean functions.

Ensembles of Boolean functions The important parameter for the ensemble is the average *sensitivity* (Eq. 2.4).

Here we specify π as the maximum entropy ensemble under a given average sensitivity

$$\pi_\lambda(f) \propto \exp(\lambda s(f)) \tag{5.3}$$

with appropriate normalization and λ as a free parameter. For $\lambda > 0$, average sensitivity $\langle s \rangle > 1$ is obtained for $K \leq 2$.

An alternative and more commonly used ensemble is

$$\pi_p(f) = p^{c_1(f)}(1-p)^{c_0(f)} \tag{5.4}$$

with $c_b(f)$ being the number of Boolean vectors x with $f(x) = b$, $b \in \{0, 1\}$. This amounts to a random generation of Boolean functions by independently placing rule table entries, a 1 occurring with probability p. For $K = 2$ and average sensitivity $\langle s \rangle = 1$, the two ensembles coincide (see appendix A.1).

Transfer functions For the results in Figures 5.3 and 5.4, the standard transfer function $\tilde{f} = f \circ \Theta$ is used: each continuous state is mapped to a Boolean value separately by applying a threshold at 1/2. Then the node's Boolean function is applied to decide if its own state value would rise towards 1 or falls towards 0, cf. Equation 2.9.

The alternative transfer function $\tilde{f}$, used to obtain results in Figure 5.5, is defined by

$$\tilde{f}_i(y) = \Theta(h_i(y)) \tag{5.5}$$

with

$$h_i(y) = a y_j y_k + b_1 y_j + b_2 y_k + c \tag{5.6}$$

for node i taking inputs from nodes j and k. The parameters a, b_1, b_2, c are chosen such that $h_i(y) = f_i(y)$ for $y_j, y_k \in \{0, 1\}$. If, for instance, f_i is an AND then $a = 1$ and $b_1 = b_2 = c = 0$ so $\tilde{f}_i(y) = 1$ if and only if $y_j y_k \geq 1/2$.

Integration Equation2.9 is integrated with the standard Euler method using a time increment $\Delta t = 10^{-4}$ and cross-checking the results with $\Delta t = 10^{-5}$. Integration is speeded up by exploiting the piece-wise linearity of the solution. At any time t, a possibly large time horizon $t_{\text{bound}}(t)$ is calculated such that all target values $\tilde{f}(t' - 1)$ are constant on $[t, t_{\text{bound}}(t)[$. Then the state values are calculated at time $t_{\text{bound}}(t)$ and the step is iterated (see appendix A.3).

Perturbations Given a map f, the evolution of states is uniquely determined by Eq. 2.9 by an initial condition $y(t)$ on a time interval of unit length, here taken as $[-1, 0] =: I$. We restrict ourselves to initial conditions that do not vary on I, $y(t) = y(0)$ for all $t \in I$. An

initial condition with a *small* perturbation is generated as

$$y'_i(t) := \epsilon_i(1 - y_i(t)) + (1 - \epsilon_i)y_i(t) \tag{5.7}$$

for $t \in I$. The perturbation amplitudes are arbitrary numbers $\epsilon_i \in]0, 1/2[$. An initial condition with a *flip* perturbation is generated as

$$y^!_i(t) := \begin{cases} 1 - y_i(t) & \text{if } i = l \\ y_i(t) & \text{otherwise} \end{cases} \tag{5.8}$$

for $t \in I$ and an arbitrary node $l \in \{1, \ldots, N\}$. Note that the total amplitude $\sum_i \epsilon_i$ of a small perturbation may exceed the unit amplitude of a flip perturbation. A small perturbation produces small deviations from the original state potentially at each node. A flip perturbation concentrates a maximal deviation at a single node.

We say that the system *heals* from the perturbation if the dynamics from perturbed and unperturbed initial condition eventually become the same except for an arbitrary time lag. Formally, healing from a small perturbation means that there are $t_0 > 0$ and $\tau > -t_0$ such that

$$y(t) = y'(t + \tau) \tag{5.9}$$

for all $t \geq t_0$. Healing from a flip perturbation means that Eq. 5.9 holds analogously for $y^!$ instead of y'. We define the heal time t_{heal} as the smallest time t_0 for which this holds.

CHAPTER 6

Stabilizing of Boolean Dynamics

"Preserve gains and maintain stability. Modesty brings gain, arrogance yields loss."

An essential aspect of Chinese philosophy in Tai chi chuan art.

In this chapter, we investigate the effect of distributing delays on the stability of Boolean dynamics in order, disorder and critical regimes. To demonstrate this point, we study the time evolution of the cumulative Hamming distance. We show that ensembles of random Boolean networks updated synchronously according to the nodes' flat distributed delays could present higher dynamical robustness against flip perturbations in critical, chaotic and frozen networks with proper distributions of delays.

6.1 Introduction

The concept of *stability* plays a central role for understanding of gene regulatory systems. One asks if two identical copies of the system being in states at moderate distance now will increase the distance in the next time step. Distance between state vectors x and y is measured by

$$d(x, y) = \sum_{i=1}^{N} |x_i - y_i| \tag{6.1}$$

called Hamming distance. Figure 2.6 illustrates the scenario for randomly generated Boolean networks by a so-called Derrida plot [50]. The expected Hamming distance of two systems at time $t + 1$ is plotted as a function of the Hamming distance at time t, averaging uniformly

over state pairs at the given distance.

The crucial quantity determining stability or instability in random Boolean networks is the sensitivity $\langle s \rangle$ averaged over the Boolean functions f_i of all nodes i. The sensitivity of a Boolean function measures the tendency of the output value to be modified by a change in the input vector (see methods for details). A change of behaviour is observed at $\langle s \rangle = 1$. The dynamics amplifies small Hamming distances when $\langle s \rangle > 1$ so adjacent trajectories separate, amounting to instability. At $\langle s \rangle < 1$, nearby trajectories tend to get closer, indicating stability. In the limit of large networks, a sharp transition from stability to instability is observed [48, 55] at $\langle s \rangle = 1$.

Taking this observation as generic would imply that regulatory networks of sufficiently sensitive switches are inherently unstable. Results for the Boolean dynamics of Eq. 1.5, however, do not allow for this general conclusion. Various other scenarios have been studied such as threshold functions instead of general Boolean functions [91], modified update schedules [35] and different types of perturbations [31]. Here we investigate the role of distributed delays for the stability of random Boolean networks.

6.2 Boolean Networks with Distributed Delays

Beside the discreteness of time and state variables, the standard Boolean dynamics makes the simplifying assumption of homogeneous transmission delay, as indicated by evaluating the state vector at time $t - 1$ in Eq. 1.5. The system operates in a clocked fashion where each node takes unit time to respond to a changing input. A real biochemical system, however, involves processes at various time scales. The time passing between production of a regulating molecule and its binding to a target site depends both on the molecule and its target site.

Therefore we study the dynamics

$$x_i(t) = f_i(x_1(t - \tau_{i1}), x_2(t - \tau_{i2}), \ldots, x_N(t - \tau_{iN})) \tag{6.2}$$

allowing a different delay τ_{ij} for each interaction between a source node j and a target node i. When generating a random Boolean network, τ_{ij} is drawn as a random integer from a flat distribution on $\{1, 2\bar{\tau} - 1\}$ for each pair of nodes i and j. The average delay $\bar{\tau}$ is a tunable parameter. The dynamics with homogeneous unit delays of Eq. 1.5 is reproduced as $\bar{\tau} = 1$.

6.3 Cumulated Hamming Distance

Let us observe the systems with distributed delays under a flip perturbation prepared as follows. For the unperturbed dynamics, we draw a state vector $x(0) \in \{0, 1\}^n$ uniformly. In order to cope with arbitrary delay values, we extend this initial condition to all earlier times, so $x(t) = x(0)$ for all $t < 0$. Then an identical Boolean network is prepared with a perturbed

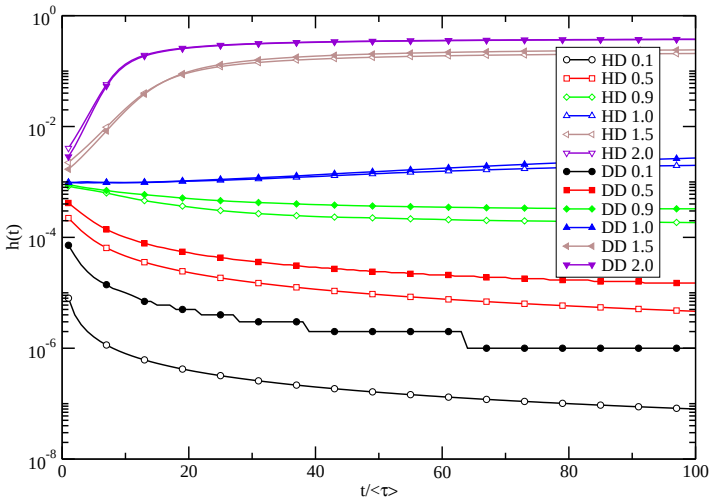


Figure 6.1: Time evolution of cumulated Hamming distances for networks with homogeneous unit delay, HD, $\bar{\tau} = 1$ (open symbols) and networks with distributed delays , DD,($\tau_{ij} \in [1, 2\bar{\tau} - 1]$) when $\bar{\tau} = 10$ (filled symbols). Symbols distinguish values of the average sensitivity (see figure legend). Each data point is an average over 1000 independent realizations of random Boolean networks (1000,2), initial conditions and perturbations.

initial condition. We draw a node i^* uniformly at random and set $y_i(t) = x_i(t)$ if and only if $i \neq i^*$ for all nodes i and times $t \leq 0$. Thus the perturbed initial condition differs from the unperturbed one at exactly one node i^*.

In order to observe the divergence or convergence of the two systems' trajectories, it is convenient to consider the following normalized *cumulated Hamming distance*

$$h(t) = (nt)^{-1} \sum_{s=1}^{t} d(x(s), y(s)) \ . \tag{6.3}$$

6.4 Results and Discussion

A random Boolean network ensemble is generated by drawing for each node two inputs $j(i)$ and $k(i)$ at random (flat distribution) and a Boolean function f_i out of the 16 Boolean functions depending on at most two inputs. The probability for function f_i is $\pi_\lambda(f_i) \propto \exp(\lambda s(f_i))$ with appropriate normalization. The distribution π is the maximum entropy ensemble under a given average sensitivity with a free parameter λ (see appendix A.1).

Figure 6.1 shows that cumulative Hamming distance of the given random Boolean ensemble with homogeneous distribution (all $\tau_{ij} = 1$) of delays decrease faster than an exponential decay in order phase with average sensitivity less than 1. In contrast it increases in a logarithmic fashion in critical regime with ($\langle s \rangle = 1$) and extremely faster than a logarithmic growth in ensembles with ($\langle s \rangle >= 1$). Distributing delays ($\tau_{ij} \in [1, 2\bar{\tau} - 1]$) with $\bar{\tau} = 10$ leads dynamics to higher rate of cumulative Hamming distance in all three phases. This could give rise to less

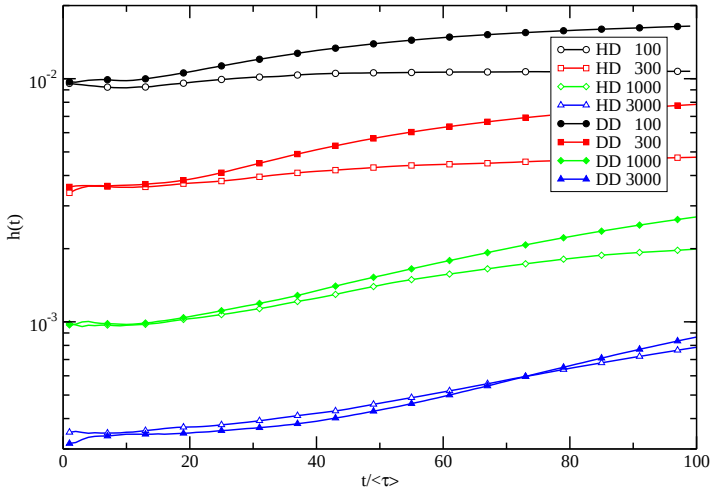


Figure 6.2: Cumulated Hamming distance versus time for 1000 random critical Boolean networks, where $\langle s \rangle = 1$ and $K = 2$ and system sizes vary from 100 to 300, and from 1000 to 3000. Open symbols show the homogeneous unit delays dynamics, HD, $\bar{\tau} = 1$, while filled symbols represent distributed delays, DD, dynamics ($\tau_{ij} \in [1, 2\bar{\tau} - 1]$) with $\bar{\tau} = 10$.

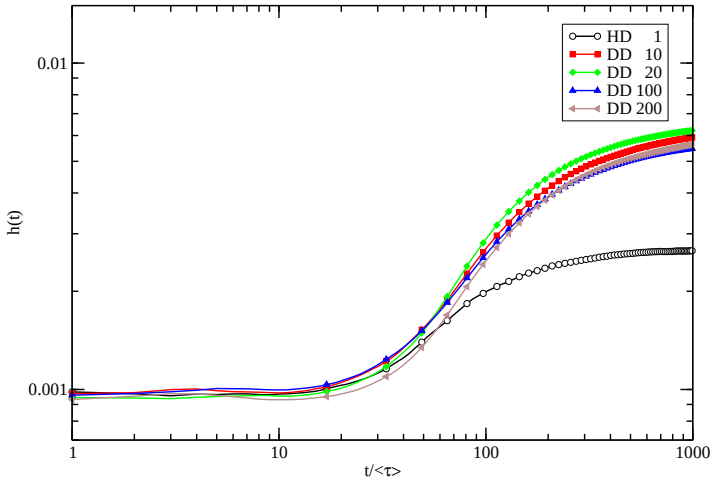


Figure 6.3: Effect of changing the average of delays on stability of critical Boolean networks. Each data point is an average over 1000 random Boolean networks (1000,2) with $\langle s \rangle = 1$. Symbols ($\circ$, $\square$, $\diamond$, $\triangle$, $\triangleleft$) refer, respectively, to distinct $\bar{\tau} = 1$, 10, 20, 100, 200.

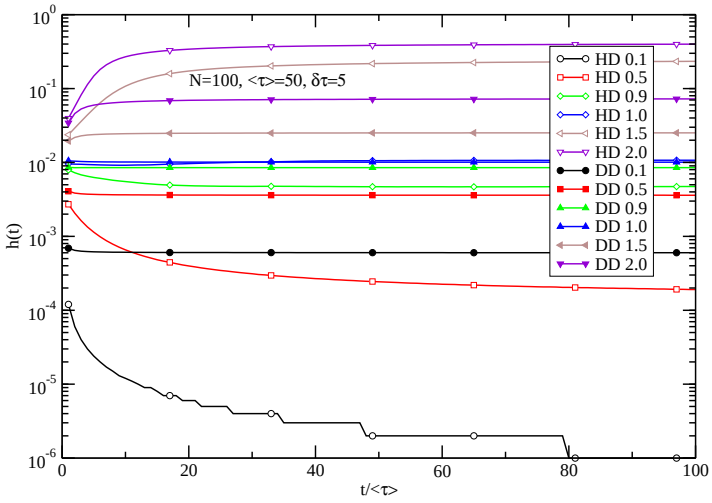


Figure 6.4: Time evolution of cumulated Hamming distances for random Boolean networks (100,2) with (open symbols) homogeneous unit delay, HD, $\bar{\tau} = 1$ and (filled symbols) distributed delays, DD, ($\tau_{ij} \in [\bar{\tau} - \delta\tau, \bar{\tau} + \delta\tau]$) with different average sensitivities, where $\bar{\tau} = 50$ and $\delta\tau = 5$. Each data point is an average over 1000 independent realizations.

dynamical robustness of the system. Note that this increasing rate is smoother in ensembles with greater average sensitivity.

Figure 6.2 demonstrates that system size does not significantly affect our analysis in critical random Boolean networks. Fixing the system size at 1000, we probed the role of stretching range of delays $\bar{\tau}$ from 10 to 20 as well as 100 to 200 on the behavior of the dynamics. Thus figure 6.3 shows that changing average of delays $\bar{\tau}$ could not reproduce more robust dynamics.

Now let us change the range of delays by adding another parameter. Again we generate random delays with a flat distribution but in different intervals around the average delay ($\tau_{ij} \in [\bar{\tau} - \delta\tau, \bar{\tau} + \delta\tau]$), where the average delay $\bar{\tau}$ and $\delta\tau$ are the tunable parameters. Here the dynamics with homogeneous unit delays of Eq. 1.5 is reproducible with $\bar{\tau} = 1$ and $\delta\tau = 0$, and the studied distribution can be obtained with $\delta\tau = \bar{\tau} - 1$ too. Thus figure 6.4 indicates the same behaviour as figure 6.1. Nevertheless Fig. 6.5 shows that distributing delays could present higher dynamical robustness against flip perturbations in critical, chaotic and frozen networks with proper values for the parameter $\delta\tau$.

In summary, distributing delays in random Boolean networks could result in differences in the stability behavior of the system. As we have shown here, these differences are captured well by the time evolution of normalized cumulated Hamming distance. An important implication of these results could help modellers understand better the higher robustness of the empirical networks than what we expected from the predictions of corresponding mathematical formalisms. Future investigation of this idea is needed to establish a more reasonable scenario as to how to distribute the delays to catch higher robustness regarding real time scales of the chemical reactions in gene regulatory systems.

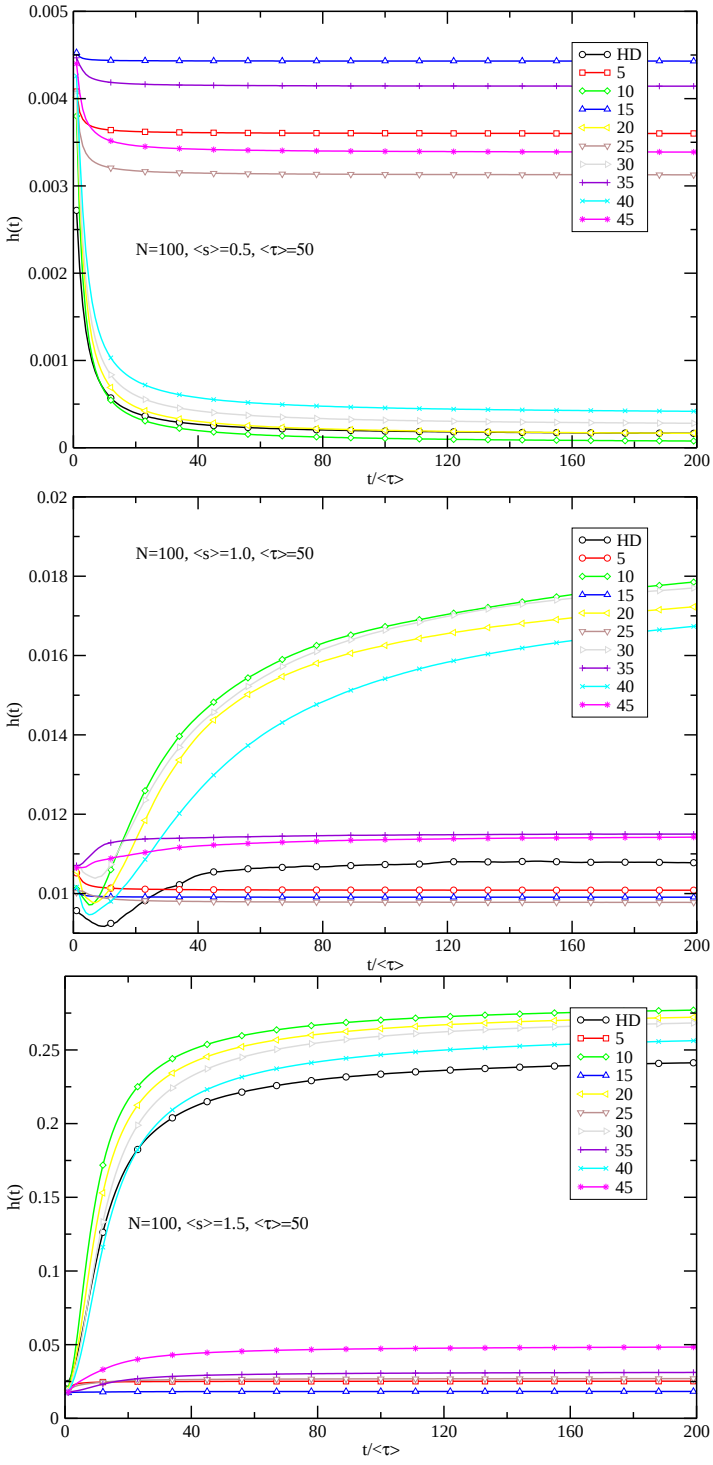


Figure 6.5: Stabilizing of Boolean dynamics by changing $\delta\tau$. Delays are distributed in a unit fashion ($\tau_{ij} \in [\bar{\tau} - \delta\tau, \bar{\tau} + \delta\tau]$), where $\bar{\tau} = 50$ and $\delta\tau$ varies from 5 to 45. Top, middle and bottom charts differ in terms of order, critical and chaotic phases respectively. Values are averaged over 1000 independent realizations of random Boolean networks (100,2), initial conditions and perturbations.

CHAPTER 7

Summary and Outlook

Investigating the influence and role of noise, fluctuations and in general perturbations on the gene regulatory dynamics not only helps us improve our models for better descriptions and more precise predictive power, but also could extend our understanding of significant concepts like complexity and evolution as well as practical issues such as cancer treatment. Hence in this dissertation we focused on this crucial subject. Our studies went in three directions.

First we addressed the role of individuals on spreading of a perturbation. We showed that, even in random Boolean networks, nodes exhibit significant differences in dynamical impact; And these differences are captured well by local and global centrality measures that can be calculated efficiently. The locally defined measures of strength and out-degree are the best predictors for the short term spreading of a perturbation from a given node. For long term predictions, principal eigenvectors of the adjacency and activity matrices perform best. These centralities result from analytic considerations of a linearized spreading scenario. Eigenvectors naturally arise as the asymptotic result of taking arbitrarily large powers of the matrix underlying the linearized dynamics.

Another direction of our work was probing the dynamical stability behaviour of random Boolean networks against two types of perturbations. We have shown that flip perturbations make a disagreement with definition of stability in corresponding continuous case. Thus it could not present a consistent stability criterion when we turn from continuous models to discrete ones. On the other hand, small perturbations provide a fresh look at the order-disorder transition in Boolean networks.

In the last part of our work, we distributed the delays to study how much differently the dynamics could behave in comparison to the models with homogeneous transmission delays. We showed that proper distributions of delays could produce more stability against

flip perturbations in critical as well as sub- and super-critical networks.

The future research should be focused on addressing the following question. What happens when mapping these three approaches, i.e. distributed delays, small perturbations and ranking of nodes then? In other words, how would be the centrality of individuals based on spreading of small perturbations in a distributed delays system?

Furthermore, one should investigate where and how the resilience of reversible Boolean functions against perturbations break down. Also one should study the dynamical behaviour in often "digital" information processing systems, for instance cellular automata, to show where the discrete approximation breaks down in terms of stability.

In addition, one should also study the possible relevance of an optimized distribution of delays with time scales of the chemical reactions in the dynamics.

APPENDIX A

Algorithms

A.1 Function Probabilities in Maximum Entropy Ensemble

The following subroutine is one that calculates the function probabilities (fpr) in maximum entropy ensembles with parameter λ, where **pot[i]** is 2^i, **fwidth** is number of possible Boolean functions (2^{2^K}) with K inputs and **mean/norm** is the given average sensitivity $\langle s \rangle$.

```
double calc_fprob(int K, int fwidth, double lambda, double *fpr)
{
  int i,j,l,s,f;
  double norm,mean;  norm=mean=0.0;
  for (f=0; f<fwidth; f++)
  {
    s=0;
    for (i=0; i<pot[K]; i++)
      for (l=1; l<pot[K]; l*=2)
        if ( ((f&pot[i])&&!(f&(pot[i^l]))) || ((f&(pot[i^l]))&&!(f&pot[i])))  s
            ++;

    norm+=fpr[f]=exp(lambda*s/pot[K]);
    mean+=fpr[f]*s/pot[K];
  }
  for (f=0; f<fwidth; f++)
     {fpr[f]/=norm;}
  return mean/norm;
}
```

codes/RBN.cpp

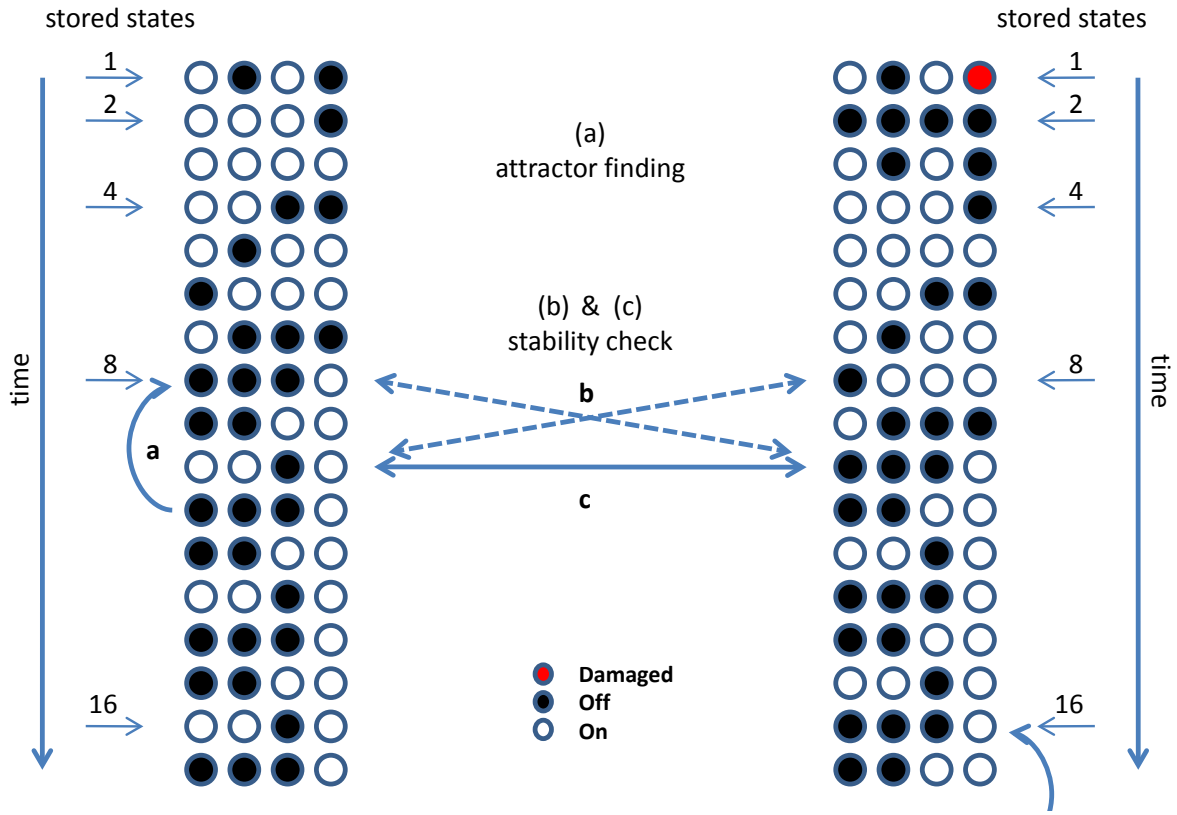


Figure A.1: Algorithm for finding a Boolean attractor and testing its stability. When two state vectors are equal in (a) comparison, the attractor of dynamics is found. Likewise equality of state vectors in (b) or (c) represents the stability of the dynamics. If both dynamics find their own attractors without any matching in (b) or (c), the dynamics is unstable.

A.2 Boolean Attractors

Let us here explain the algorithm used in finding attractors of a given Boolean network. Starting from an initial state vector, the vector is stored whenever the updating time is a power of 2. Then in any time step the updated state vector is compared with the last stored vector. If these two are exactly same, the attractor is found, Fig. A.1 (a). Likewise one could check the stability of an attractor, Fig. A.1 (b) and (c). When the state vectors of original dynamics and the perturbed one are equal in the same time step, Fig. A.1 (c), attractor is stable. Similarly an attractor is also stable whenever one of the state vectors of original and perturbed dynamics is equal to the stored state vector of the other one; in this case perturbation has only caused a phase shift between the two attractors, Fig. A.1 (b). Therefore an attractor is unstable when two dynamics end at two different attractors.

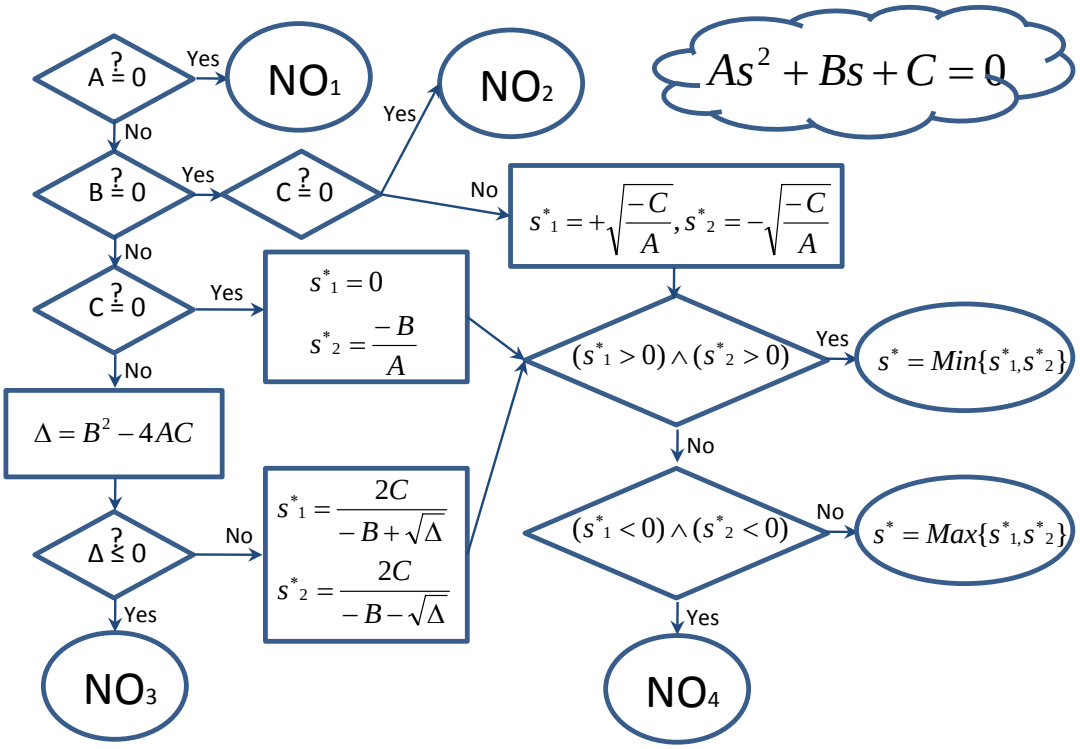


Figure A.2: Solution of quadratic equation. Since the largest solution with $0 < s^* < 1$ is our target, zero, negative, double and complex roots are discarded, NO_is.

A.3 Continuous Dynamics

Let us mention two major issues we faced while simulating the continuous dynamics. The first issue concerns expediting the integration of Eq. 2.9, where similar to the case of discrete molecular dynamics, the dynamics is updated in a discrete fashion by δt in which δt does not always have the same value. It presents the closest time at which the first event happens in the dynamics, either a node's inputs (or more than one node simultaneously) pass the threshold θ ($\{\min\{t_i^*\}\}$) or the target value of a node (or a couple of nodes) changes ($\min\{t^*_{i(stored)} + 1\}$). This means $\delta t = \{\min\{\min\{\delta t^*_{i(stored)} + 1\}, \min\{\delta t_i^*\}\}$. Let us now calculate the threshold time δt_i^* for nodes with two inputs. Consider a node (i) receiving input from two other nodes (x_1 and x_2). The effective input is a polynomial in the two input values

$$\dot{s_i}(t) = \alpha[f(h_i(t-1)) - s_i(t)] \quad f : R \longrightarrow [0,1] \tag{A.1}$$

$$h_i(t) = ax_1(t)x_2(t) + b_1x_1(t) + b_2x_2(t) + c \tag{A.2}$$

where, for instance, a node performing $(x_1 \wedge \bar{x_2})$ has $a = -1$, $b_1 = 1$ and $b_2 = c = 0$. Fixing initial conditions $x_i(0) = x_i^0$ and target values x_i^∞, $i \in \{1, 2\}$, the input values follow the time evolution

$$x_i(t) = (x_i^0 - x_i^\infty) \exp(-\alpha(t - t^0)) + x_i^\infty \tag{A.3}$$

as long as target values remain the same. Using a transformed time variable $s = \exp(-\alpha t)$, this reads

$$x_i(s) = \xi_i s + x_i^\infty \tag{A.4}$$

with $\xi_i = x_i^0 - x_i^\infty$. Inserting into Eq. A.2 we obtain

$$\begin{aligned} h(s) &= a(\xi_1 s + x_1^\infty)(\xi_2 s + x_2^\infty) + b_1(\xi_1 s + x_1^\infty) + b_2[\xi_2 s + x_2^\infty] + c \\ &= a\xi_1\xi_2 s^2 + [a(\xi_1 x_2^\infty + \xi_2 x_1^\infty) + b_1\xi_1 + b_2\xi_2]s^1 + [b_1 x_1^\infty + b_2 x_2^\infty + c + a x_1^\infty x_2^\infty]s^0 \end{aligned} \tag{A.5}$$

as a second order polynomial in s. Now the next (future) threshold crossing time t^* is obtained by solving the second order equation

$$h(s^*) = \theta \tag{A.6}$$

where θ is the threshold of the node under consideration. The largest solution with $0 < s^* < 1$ transforms back into the desired t^* as

$$\delta t_i^* = t^* - t^0 = -\alpha^{-1} \ln s^* \ . \tag{A.7}$$

Figure A.2 shows the practical algorithm for finding solutions of Equation A.6. Where A, B and C take their values from A.8.

$$\begin{aligned} A &= a\xi_1\xi_2 \\ B &= a(\xi_1 x_2^\infty + \xi_2 x_1^\infty) + b_1\xi_1 + b_2\xi_2 \\ C &= a x_1^\infty x_2^\infty + b_1 x_1^\infty + b_2 x_2^\infty + c - \theta \end{aligned} \tag{A.8}$$

To avoid the precision problem (Fig. 2.10 (e)) working with double variables in $C++$, 2.10, we rescaled the variables following equations A.9 where $\gamma = \frac{1}{\Delta t}$ represents the time increment Δt in the standard Euler method; We used the C++ *Big Integer library* to deal with larger integer variables.

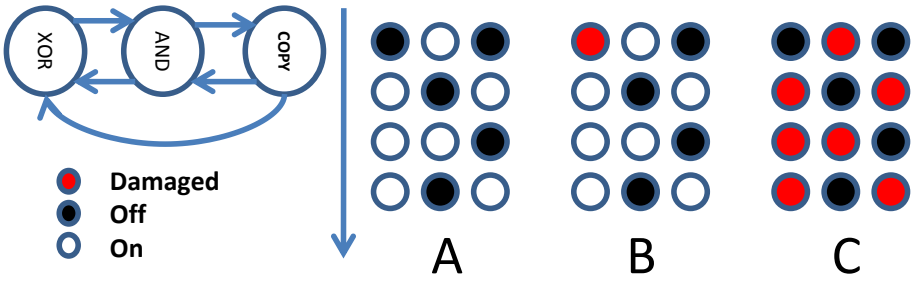


Figure A.3: Left-hand side up: A small Boolean network with three nodes is graphically presented. Right-hand side: The corresponding dynamics. The vertical arrow shows ongoing discrete time; the panels A, B and C represent dynamics started from same initial states, while the state of first node in (B) and the second one in (C) is perturbed (flipped). In this case, the system can overcome the damage on the first node but not on the second. After testing more samples (140 random realizations with different initial states and damages), dynamical impact of nodes can be measured: $h(\vec{t}) = (17, 60, 63)$. Calculating the out-degree, strength and eigenvector centrality for adjacency and activity matrices leads to $\vec{d} = (1, 2, 2)$, $\vec{\sigma} = (0.5, 1.5, 2)$, $\vec{e} = (0.6, 1, 1)$, $\vec{\epsilon} = (0.3, 0.5, 0.7)$. Then predictive powers (see Sec. 4.8) are calculated by rank order correlation as follows: $\mathcal{P}_\epsilon = \mathcal{P}_\sigma = 1$ and $\mathcal{P}_e = \mathcal{P}_d = 0.86$.

$$\begin{aligned} s' &= \gamma s \\ b_1' &= \gamma b_1 \\ b_2' &= \gamma b_2 \\ c' &= \gamma^2 c \\ \theta' &= \gamma^2 \theta \end{aligned} \tag{A.9}$$

The second issue we faced when simulating the continuous dynamics was finding attractors and making the comparison between unperturbed and perturbed dynamics. In practice finding continuous limit cycles is similar to the Boolean counterpart (see Fig. A.1), while instead of storing the states of nodes, the events, passing the threshold times (see table 2.2), were stored for the comparisons; and events in a time interval equal to the system delay were compared.

A.4 Estimation of Dynamical Impact

Figure A.3 illustrates how dynamical impact as well as centrality measures and their predictive powers in a simple Boolean network can be estimated and calculated.

A.5 Eigenvector

Applying the iteration method eigenvector corresponding to the largest eigenvalue was calculated, see following code.

```
int calc_evc(double* c,double** M , int n)
{
  int discard_net=0;
  int i,j;
  double error; double xerror=0.0;
  int iteration=0;
  double* r = new double[n];
  do
  {
    iteration++;
    xerror=error;
    norm=0.0;
    for (j=0; j<n; j++)
      {
         r[j]=0.0;
         for (i=0; i<n; i++)
            r[j]+=c[i]*M[i][j];
         norm+=r[j]*r[j];
      }
    if ( (iteration>1000) || (norm<=1e-20)) {discard_net=1; break;}
    // check converging by checking the errors
    norm= sqrt(norm);
    error=0.0;
    for (j=0; j<n; j++)
    {
      r[j]/=norm;
      error+=(r[j]-c[j])*(r[j]-c[j]);
      if (iteration>=900) c[j]=(c[j]+r[j])/2;
      else       c[j]=r[j];
    }
  }
  while ( error/n>1e-15);
  return discard_net;
}
```

codes/Eigenvector.cpp

A.6 Ranking

By this subroutine, rank of *pos* in vector *a* is calculated.

```
double calc_rank(double *a, int pos ,int n)
```

```
// actually the rank minus the average rank N/2
{
  double ng; int i;
  for (i=ng=0; i<n; i++)
    if (a[i]>a[pos]) ng+=1.0;
    else if (a[i]==a[pos]) ng+=0.5;
  return ng-(double)n/2;
}
```

codes/Rank.cpp

List of Figures

List of Tables

Bibliography

[1] F. Jacob and J. Monod. Genetic regulatory mechanisms in the synthesis of proteins. *Journal of Molecular Biology*, 3(3):318–356, June 1961.

[2] H. De Jong. Modeling and simulation of genetic regulatory systems: A literature review. *J Comput Biol*, 9(1):67–103, 2002.

[3] B. Lewin. *Genes VII.* Oxford University Press, Oxford, 1999.

[4] B. Lewin. *Genes II.* Wiley, New York, 1983.

[5] M.W. Hahn, G.A. Wray, et al. The g-value paradox. *Evolution and Development*, 4(2):73–75, 2002.

[6] R.J. Taft, M. Pheasant, and J.S. Mattick. The relationship between non-protein-coding dna and eukaryotic complexity. *Bioessays*, 29(3):288–299, 2007.

[7] M. Ptashne and A. Gann. *Genes & signals.* CSHL Press, 2002.

[8] G.A. Wray. The evolutionary significance of cis-regulatory mutations. *Nature Reviews Genetics*, 8(3):206–216, 2007.

[9] S. De, S.A. Teichmann, and M.M. Babu. The impact of genomic neighborhood on the evolution of human and chimpanzee transcriptome. *Genome research*, 19(5):785–794, 2009.

[10] D. Hanahan and R.A. Weinberg. Hallmarks of cancer: the next generation. *Cell*, 144(5):646–674, 2011.

[11] D. Hanahan and R.A. Weinberg. The hallmarks of cancer. *cell*, 100(1):57–70, 2000.

[12] A. Pomerance, E. Ott, M. Girvan, and W. Losert. The effect of network topology on the stability of discrete state models of genetic control. *Proc Natl Acad Sci USA*, 106(20):8209–8214, 2009.

[13] S. Bornholdt. Systems Biology: Less Is More in Modeling Large Genetic Networks. *Science*, 310(5747):449–451, 2005.

[14] MH Jensen, K. Sneppen, and G. Tiana. Sustained oscillations and time delays in gene expression of protein hes1. *FEBS Lett*, 541, 2003.

[15] L. Glass. Classification of biological networks by their qualitative dynamics. *J Theor Biol*, 54(1):85–107, 1975.

[16] J. Norrell, B. Samuelsson, and J.E.S. Socolar. Attractors in continuous and boolean networks. *Phys Rev E*, 76(4):46122, 2007.

[17] P. Oliveri and E.H. Davidson. Gene regulatory network controlling embryonic specification in the sea urchin. *Current opinion in genetics & development*, 14(4):351–360, 2004.

[18] S. A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *J Theor Biol*, 22:437–467, 1969.

[19] H. A. Carteret, K. J. Rose, and S. A. Kauffman. Maximum power efficiency and criticality in random boolean networks. *Phys. Rev. Lett.*, 101(21):218702, Nov 2008.

[20] I. Albert, J. Thakar, S. Li, R. Zhang, and R. Albert. Boolean network simulations for life scientists. *Source Code Biol Med*, 3:16, Januar 2008.

[21] B. Drossel. Random boolean networks. *Reviews of Nonlinear Dynamics and Complexity*, 1:69–110, 2007.

[22] T. Helikar, J. Konvalina, J. Heidel, and Jim A. Rogers. Emergent decision-making in biological signal transduction networks. *Proc Natl Acad Sci USA*, 105(6):1913–1918, 2008.

[23] D. Sahoo, J. Seita, D. Bhattacharya, Matthew A. Inlay, Irving L. Weissman, Sylvia K. Plevritis, and David L. Dill. MiDReG: A method of mining developmentally regulated genes using Boolean implications. *Proc Natl Acad Sci USA*, 107(13):5732–5737, 2010.

[24] R. Albert and H.G. Othmer. The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in Drosophila melanogaster. *J Theor Biol*, 223(1):1–18, 2003.

[25] F. Li, T. Long, Y. Lu, Q. Ouyang, and C. Tang. The yeast cell-cycle network is robustly designed. *Proc Natl Acad Sci USA*, 101(14):4781–4786, Apr 2004.

[26] A. Samal and S. Jain. The regulatory network of e. coli metabolism as a boolean dynamical system exhibits both homeostasis and flexibility of response. *BMC Syst Biol*, 2(1):21, 2008.

[27] R. Thomas, D. Thieffry, and M. Kaufman. Dynamical behaviour of biological regulatory networks-i. biological role of feedback loops and practical use of the concept of the loop-characteristic state. *Bulletin of mathematical biology*, 57(2):247–276, 1995.

[28] L. Mendoza and E.R. Alvarez-Buylla. Dynamics of the genetic regulatory network forarabidopsis thalianaflower morphogenesis. *Journal of theoretical biology*, 193(2):307–319, 1998.

[29] S. Bornholdt. Boolean network models of cellular regulation: prospects and limitations. *Journal of the Royal Society Interface*, 5(Suppl 1):S85–S94, 2008.

[30] F. Ghanbarnejad and K. Klemm. Impact of individual nodes in boolean network dynamics. *Arxiv preprint arXiv:1111.5334*, 2011.

[31] F. Ghanbarnejad and K. Klemm. Stability of boolean and continuous dynamics. *Phys. Rev. Lett.*, 107:188701, Oct 2011.

[32] I. Shmulevich and S.A. Kauffman. Activities and sensitivities in Boolean network models. *Phys Rev Lett*, 93(4):48701, 2004.

[33] R. Albert. Scale-free networks in cell biology. *Journal of cell science*, 118(21):4947, 2005.

[34] F. Greil. *Dynamics of Boolean networks*. Phd thesis, Technische Universität Darmstadt, may 2009.

[35] C. Gershenson. Introduction to random boolean networks. *Arxiv preprint nlin/0408006*, 2004.

[36] C. Gershenson. Updating schemes in random boolean networks: Do they really matter. In *Artificial Life IX Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems*, pages 238–243. MIT Press, 2004.

[37] T. Skodawessely and K. Klemm. finding attractors in asynchronous boolean dynamics. *Advances in Complex Systems*, 14:439–449, 2011.

[38] T. Skodawessely. *State space exploration for Boolean dynamics*. Diploma thesis, Universität Leipzig, 2010.

[39] S. Kauffman. Homeostasis and differentiation in random genetic control networks. *Nature*, 224:177–178, 1969.

[40] S. Bilke and F. Sjunnesson. Stability of the kauffman model. *Phys Rev E*, 65(1):016129, Dec 2001.

[41] J. E. S. Socolar and S. A. Kauffman. Scaling in ordered and critical random boolean networks. *Phys. Rev. Lett.*, 90:068702, Feb 2003.

[42] B. Samuelsson and C. Troein. Superpolynomial growth in the number of attractors in kauffman networks. *Phys. Rev. Lett.*, 90:098701, Mar 2003.

[43] L. Kadanoff, S. Coppersmith, and M. Aldana. Boolean dynamics with random couplings. *Arxiv preprint nlin/0204062*, 2002.

[44] T. Mihaljev and B. Drossel. Scaling in a general class of critical random boolean networks. *Phys. Rev. E*, 74:046101, Oct 2006.

[45] A. Szejka, T. Mihaljev, and B. Drossel. The phase diagram of random threshold networks. *New Journal of Physics*, 10(6):063009, 2008.

[46] M. Aldana and P. Cluzel. A natural class of robust networks. *Proceedings of the National Academy of Sciences*, 100(15):8710–8714, 2003.

[47] B. Drossel and F. Greil. Critical boolean networks with scale-free in-degree distribution. *Phys. Rev. E*, 80:026102, Aug 2009.

[48] S. A. Kauffman. *The Origins of Order*. Oxford University Press, New York, 1993.

[49] K. Klemm and S. Bornholdt. Stable and unstable attractors in boolean networks. *Phys Rev E*, 72(5):055101, Nov 2005.

[50] B. Derrida and Y. Pomeau. Random networks of automata: A simple annealed approximation. *Europhys Lett*, 1:45, 1986.

[51] I. Shmulevich, H. Lähdesmäki, Edward R. Dougherty, J. Astola, and W. Zhang. The role of certain Post classes in Boolean network models of genetic networks. *Proc Natl Acad Sci USA*, 100(19):10734–10739, 2003.

[52] C. Fretter, A. Szejka, and B. Drossel. Perturbation propagation in random and evolved boolean networks. *New Journal of Physics*, 11(3):033005, 2009.

[53] Tiago P. Peixoto. Redundancy and error resilience in boolean networks. *Phys Rev Lett*, 104(4):048701, Jan 2010.

[54] C. Schmal, Tiago P Peixoto, and B. Drossel. Boolean networks with robust and reliable trajectories. *New Journal of Physics*, 12(11):113054, 2010.

[55] C. Seshadhri, Y. Vorobeychik, Jackson R. Mayo, Robert C. Armstrong, and Joseph R. Ruthruff. Influence and dynamic behavior in random boolean networks. *Phys. Rev. Lett.*, 107:108701, Sep 2011.

[56] B. Luque and R.V. Solé. Lyapunov exponents in random boolean networks. *Physica A: Statistical Mechanics and its Applications*, 284(1):33–45, 2000.

[57] H. Kantz, T. Schreiber, and R.S. Mackay. *Nonlinear time series analysis*, volume 2000. Cambridge university press Cambridge, 1997.

[58] R. Zhang, H.L.D.S. Cavalcante, Z. Gao, D.J. Gauthier, J.E.S. Socolar, M.M. Adams, and D.P. Lathrop. Boolean chaos. *Physical Review E*, 80(4):045202, 2009.

[59] M. Davidich and S. Bornholdt. Boolean network model predicts cell cycle sequence of fission yeast. *PLoS ONE*, 3(2):e1672, 2008.

[60] M. Davidich and S. Bornholdt. The transition from differential equations to boolean networks: A case study in simplifying a regulatory network model. *J Theor Biol*, 255(3):269 – 277, 2008.

[61] Steven H Strogatz. *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry and Engineering*. Westview Press, Boulder, 1994.

[62] L. Glass and S.A. Kauffman. Co-operative components, spatial localization and oscillatory cellular dynamics. *J Theor Biol*, 34(2):219–237, 1972.

[63] S. Braunewell and S. Bornholdt. Reliability of regulatory networks and its evolution. *J Theor Biol*, 258(4):502 – 512, 2009.

[64] J. Norrell and J.E.S. Socolar. Boolean modeling of collective effects in complex networks. *Phys Rev E*, 79(6):61908, 2009.

[65] E. Gehrmann and B. Drossel. Boolean versus continuous dynamics on simple two-gene modules. *Phys Rev E*, 82(4):046120, Oct 2010.

[66] S. A. Kauffman, C. Peterson, B. Samuelsson, and C. Troein. Random Boolean network models and the yeast transcriptional network. *Proc Natl Acad Sci USA*, 100(25):14796–14799, 2003.

[67] Q. Xia, W. Liu, L.and Ye, and G. Hu. Inference of gene regulatory networks with the strong-inhibition boolean model. *New Journal of Physics*, 13(8):083002, 2011.

[68] A. Fauré, A. Naldi, C. Chaouiya, and D. Thieffry. Dynamical analysis of a generic boolean model for the control of the mammalian cell cycle. *Bioinformatics*, 22(14):e124, 2006.

[69] G. Boldhaus, N. Bertschinger, J. Rauh, E. Olbrich, and K. Klemm. Robustness of boolean dynamics under knockouts. *Physical Review E*, 82(2):021916, 2010.

[70] L. Mendoza and I. Xenarios. A method for the generation of standardized qualitative dynamical systems of regulatory networks. *Theoretical biology and medical modelling*, 3(1):13, 2006.

[71] P. Rue, A. J. Pons, N. Domedel-Puig, and J. Garcia-Ojalvo. Relaxation dynamics and frequency response of a noisy cell signaling network. *Chaos*, 20, 2010.

[72] A. Eldar and M. B. Eolwitz. Functional roles for noise in genetic circuits. *Nature*, 467:167–173, 2010.

[73] K. Klemm and S. Bornholdt. Topology of biological networks and reliability of information processing. *Proc Natl Acad Sci USA*, 102(51):18414–18419, 2005.

[74] S. Braunewell and S. Bornholdt. Superstability of the yeast cell-cycle dynamics: Ensuring causality in the presence of biochemical stochasticity. *J Theor Biol*, 245(4):638 – 643, 2007.

[75] T.P. Peixoto and B. Drossel. Noise in random boolean networks. *Physical Review E*, 79(3):036108, 2009.

[76] I. Lestas, G. Vinnicombe, and J. Paulsson. Fundamental limits on the suppression of molecular fluctuations. *Nature*, 467:174–178, 2010.

[77] A.L. Barabasi H. Jeong, S. P. Mason and Z. N. Oltvai. Lethality and centrality in protein networks. *Nature*, 411:41–42, 2001.

[78] M. Kitsak, L.K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H.E. Stanley, and H.A. Makse. Identification of influential spreaders in complex networks. *Nature Physics*, 2010.

[79] B. Bollobás. The evolution of sparse graphs. In B. Bollobás, editor, *Graph Theory and Combinatorics*, pages 35–37. Academic Press (London), 1984.

[80] L.C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41, 1977.

[81] P. Bonacich. Factoring and weighting approaches to status scores and clique identification. *Journal of Mathematical Sociology*, 2:113–120, 1972.

[82] M.E.J. Newman. *Networks: an introduction*. Oxford Univ Press, 2010.

[83] K. Klemm, M. Serrano, V.M. Eguiluz, and M.S. Miguel. A measure of individual role in collective dynamics. *Scientific Reports*, 2(292), 2012.

[84] S. Wuchty and Peter F. Stadler. Centers of complex networks. *Journal of Theoretical Biology*, 223:45–53, 2003.

[85] A.A. Moreira and L.A.N. Amaral. Canalizing kauffman networks: Nonergodicity and its effect on their critical behavior. *Physical review letters*, 94(21):218702, 2005.

[86] S. A. Kauffman, C. Peterson, B. Samuelsson, and C. Troein. Genetic networks with canalyzing Boolean rules are always stable. *Proc Natl Acad Sci USA*, 101(49):17102–17107, 2004.

[87] T. Rohlf, N. Gulbahce, and C. Teuscher. Damage spreading and criticality in finite random dynamical networks. *Phys Rev Lett*, 99(24):248701, Dec 2007.

[88] C. G. Langton. Computation at the edge of chaos: Phase transitions and emergent computation. *Physica D*, 42(1-3):12 – 37, 1990.

[89] Y. Li, Z.R. Liu, and J.B. Zhang. Dynamics of network motifs in genetic regulatory networks. *Chinese Physics*, 16:2587–2594, 2007.

[90] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824, 2002.

[91] T. Rohlf and S. Bornholdt. Criticality in random threshold networks: annealed approximation and beyond. *Physica A: Statistical Mechanics and its Applications*, 310(1):245–259, 2002.

Selbständigkeitserklärung

Hiermit erkläre ich, die vorliegende Dissertation selbständig und ohne unzulässige fremde Hilfe angefertigt zu haben. Ich habe keine anderen als die angeführten Quellen und Hilfsmittel benutzt und sämtliche Textstellen, die wörtlich oder sinngemäß aus veröffentlichten oder unveröffentlichten Schriften entnommen wurden, und alle Angaben, die auf mündlichen Auskünften beruhen, als solche kenntlich gemacht. Ebenfalls sind alle von anderen Personen bereitgestellten Materialien oder erbrachten Dienstleistungen als solche gekennzeichnet.

Leipzig, den 3. Mai 2012

(Fakhteh Ghanbarnejad)